

Note: You can click a cross-reference entry to electronically turn to that page.

Inmagic[®] DB/Text[®] *WebPublisher* User's Manual

Publishing Textbases on the Web



INMAGIC[®]

**A guide for using DB/Text *WebPublisher*
and the DB/TextWorks[®] buildware**

Copyright © 2001 by Inmagic, Inc. All rights reserved.

Inmagic®, the Inmagic logo, DB/Text®, DB/TextWorks®, DB/SearchWorks®, BiblioTech®, and BiblioTech PRO® are registered trademarks, and Inmagic.net™, BibSpeed™, and PowerPack™ are trademarks of Inmagic, Inc. Other brand and product names are trademarks or registered trademarks of their respective holders.

The information in this document is subject to change without notice and should not be construed as a commitment by Inmagic, Inc., which assumes no responsibility for any errors that may appear in this document. Use of any other product name does not imply endorsement of that product by Inmagic, Inc.

WARRANTY

INMAGIC, INC. MAKES NO WARRANTY, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY AND FITNESS. INMAGIC, INC. SHALL NOT BE LIABLE FOR ANY LOST PROFITS OR ANY OTHER INCIDENTAL OR CONSEQUENTIAL DAMAGES. IN PARTICULAR, INMAGIC, INC. SHALL HAVE NO LIABILITY FOR ANY DATA OR PROGRAMS STORED OR USED WITH THIS PRODUCT, INCLUDING THE COSTS OF RECOVERING SUCH PROGRAMS OR DATA.

U.S. GOVERNMENT: If Licensee is acquiring the software on behalf of any unit or agency of the U.S. Government, the following shall apply:

(a) For units of the Department of Defense: RESTRICTED RIGHTS LEGEND: Use, duplication or disclosure by the Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data Clause at DFARS 252.227-7013. (b) For any other unit or agency: NOTICE - Notwithstanding any other lease or license agreement that may pertain to, or accompanying the delivery of, the computer software and accompanying documentation, the rights of the Government regarding its use, reproduction and disclosure are as set forth in Clause 52.227-19(c)(2) of the FAR.

Contractor/Manufacturer is Inmagic, Inc., 800 West Cummings Park, Woburn, MA 01801-6357, U.S.A.

Questions regarding any Inmagic product should be addressed to Inmagic, Inc. or your local Inmagic dealer.

Inmagic, Inc., 800 West Cummings Park, Woburn, MA 01801-6357 U.S.A.
Telephone: 781-938-4442 or 800-229-8398
Fax: 781-938-6393
EMail: inmagic@inmagic.com
<http://www.inmagic.com>

Note: If you have a Web browser, you can click the URL to go to the Inmagic Web site.

Contents

Chapter 1: Installation	1
Installation Instructions	1
Permissions	7
Creating a Virtual Directory	8
Chapter 2: Fundamentals	13
Documentation and Other Resources	13
Terminology	14
Learning about DB/Text <i>WebPublisher</i>	14
Chapter 3: Using the DB/TextWorks Buildware	23
Textbases	23
Forms	26
Query Screens	42
Menu Screens	54
Images	59
Intranets	61
Testing your Textbases: A Checklist	64
Chapter 4: Client Requirements and Resources	65
Client Requirements	65
How Clients Get Help	65
Query Syntax	66
Chapter 5: For the Webmaster	67
Publishing a DB/TextWorks Textbase	67
Publishing a “Live” Textbase	68
Publishing on an Intranet	68
File Locations	69
Initialization Files	72
Query Sets	73
Chapter 6: Tutorials	75
Exercise #1: Searching on the Web	75
Exercise #2: Using the Buildware	79

Note: You can click a Table of Contents entry to electronically turn to that page.

Appendix: Troubleshooting.....	85
Glossary.....	89
Index.....	95

Chapter 1: Installation

This manual explains how to use Inmagic® DB/Text® *WebPublisher* together with Inmagic DB/TextWorks® to publish textbases on the Web.

- **DB/Text *WebPublisher*** is a Web publishing and information retrieval tool, which accepts queries from standard Web browsers, such as Microsoft Internet Explorer and Netscape Navigator, and returns search results in dynamically generated HTML reports.
- **DB/TextWorks** is the “buildware,” the database software that you use to build the textbases (text databases) that you want to publish.

Although they are separate applications, the two products are closely related, and you need both of them to publish information on the Internet or an intranet.

Installation Instructions

To use DB/Text *WebPublisher*, you need to install two separate applications: DB/TextWorks and DB/Text *WebPublisher*. It does not matter which one you install first. Note that you probably will install them on separate machines, although that is not a requirement.

In addition, you must have HTTP server software, such as Internet Information Services (IIS), which is the free Windows Web server software available for Microsoft Windows NT version 4.0 or later systems.

For DB/Text *WebPublisher* to work with your server software, you must properly configure permissions (access rights) for both the *WebPublisher* installation directory and the virtual directory (also known as an alias) that you create on your server. These permissions are explained in “Permissions” on page 7.

Compatibility

For best results, use the latest available versions of both *WebPublisher* and DB/TextWorks. If you have upgraded from *WebPublisher* version 2.2b or earlier, use DB/TextWorks version 4 or later to re-export to HTML any query or menu screens created for the previous version of *WebPublisher*.

See the Inmagic Web site (www.inmagic.com) for recent product updates. Those enrolled in the InmagicADVANTAGE maintenance program can download new product releases directly from the Inmagic Web site on a regular basis.

Note: If you have a Web browser, you can click the URL to go to the Inmagic Web site.

Installing DB/TextWorks

DB/TextWorks is the buildware that you use to create and edit textbases, forms, and other components of the searchable textbases that you want to publish on the Web. Only the textbase designer should have access to DB/TextWorks. Web clients should not have access to the DB/TextWorks software, except possibly on an intranet.

Install DB/TextWorks on a machine that you will be able to access frequently, such as any local PC running 32-bit versions of Windows (for example, 95, 98, ME, NT, 2000, and so forth). Ideally, this PC will have a network connection to the server where *WebPublisher* is installed. If you prefer, you can install DB/TextWorks on the Windows NT server hosting *WebPublisher*, in any directory that is protected from clients. To install DB/TextWorks, follow the installation instructions provided with that software. Basically, you insert the DB/TextWorks CD-ROM in the drive and run Setup.

Installing DB/Text *WebPublisher*

To use DB/Text *WebPublisher* on the Web, you must install the software and then map its installation directory to a virtual directory (also known as an alias). Doing this enables *WebPublisher* to construct URLs for accessing information in the installation directory structure. The virtual directory becomes the equivalent of the installation directory.

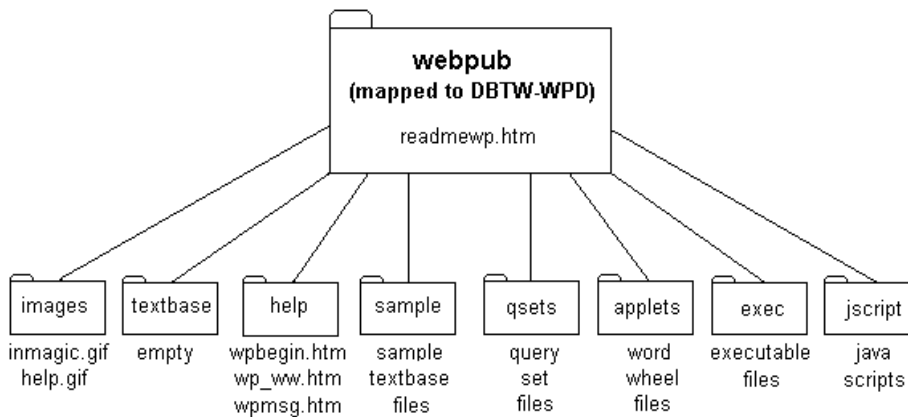
Important! Permissions must be set properly for *WebPublisher* to work on the Web. Almost all *WebPublisher* problems are caused by incorrect configuration or permission settings. For more information, see “Permissions” on page 7.

Install DB/Text *WebPublisher* on a server that is running Microsoft Windows NT Server version 4.0 or later, or Windows 2000 Server. *WebPublisher* can also run under Windows systems (such as, 95, 98, ME, NT Workstation version 4, or 2000 Professional), although this is not recommended for production use. The server should have at least 16MB of RAM (32MB or more preferred) available for *WebPublisher* and at least 1.6MB of disk space free (2MB preferred). It must have HTTP server software and TCP/IP installed and running. The server needs an IP address (for example, 192.9.200.4). If the server is connected to the Internet, the IP address needs to be registered. *WebPublisher* supports ISAPI and CGI.

Tip! If you are using Microsoft Internet Information Services (IIS), we recommend ISAPI. If you are using server software other than IIS, we recommend CGI (the industry standard).

WebPublisher Directory Structure

DB/Text *WebPublisher* uses the directory structure illustrated below. Additional subdirectory levels are not shown.



To install DB/Text *WebPublisher*

1. On the HTTP server, log in with Administrator privileges.

Note: If you plan to install over an existing copy of *WebPublisher*, you must stop your HTTP server World Wide Web service.

2. Insert the DB/Text *WebPublisher* Setup CD-ROM in the drive. If necessary, use the Windows Run command on the Start menu to run the Setup program. For example, type D:\SETUP and click **OK**.
3. When the Setup program starts, follow the instructions on your screen to accept the license agreement and to enter your registration information.
4. When the Select Components dialog box appears, specify a destination directory on the HTTP server. Later, you will map the specified destination directory to a virtual server directory called /DBTW-WPD.

Note: If you want to be able perform the tutorials in Chapter 6, be sure to install the sample textbase files.

5. Click **Next**.

6. In the Specify Textbase Search Paths dialog box, specify one or more locations where you want *WebPublisher* to search for the textbases you will publish. When necessary, *WebPublisher* can recursively search all subdirectories under those locations when a query is submitted.

Use a comma to separate multiple locations. If you specify a machine other than the HTTP server machine, use the Universal Naming Convention (UNC) directory name instead of a drive letter. For example:

`C:\WEBPUB\TEXTBASE\, C:\DATA, \\SERVER2\TBS`

The locations you specify are written to an initialization file called DBTWPub.INI. Later, you can edit that file to add or change locations. For more information, see “Textbase File Locations” on page 69.

7. Click **Next**.
8. In the Specify Logo/Background Image URL dialog box, type the URL for a default Web directory where you will place logo and background image files used on forms.

For example, if you keep images in the MYLOGOS subdirectory under the server home directory, type `/MYLOGOS/`. The leading slash (/) makes the URL relative to the current server root. Specify the full path (for example, `http://hostname/mylogos/`) only if the file server is different from the installation server.



The URL you specify does not have to exist and will not be created automatically. It is simply a text string that *WebPublisher* will insert in front of image file names. For example, if you specify /MYLOGOS as the URL, *WebPublisher* will construct URLs such as `http://hostname/mylogos/acme.gif`.

Note: Later, you will map the *WebPublisher* installation directory (for example, `C:\PROGRAM FILES\INMAGIC\WEBPUB`) to a virtual directory called /DBTW-WPD. If you plan to keep images in a subdirectory under the *WebPublisher* installation directory, use the virtual directory name in the URL. For example, type /DBTW-WPD/MYLOGOS.

The URL you specify will be written to an initialization file called DBTWPUB.INI. Later, you can edit that file to add or change locations. For more information, see “Logo and Background Image File Locations” on page 71.

9. Click **Next**.
10. In the Specify Textbase Image URL dialog box, type the URL for a default directory where you will place image files referenced in textbase records.

For example, if you keep textbase record images in the TBIMAGES subdirectory under the server home directory, type /TBIMAGES. The leading slash (/) makes the URL relative to the current server root.

Specify the full path (for example, `http://hostname/tbimages`) only if the file server is different from the installation server.



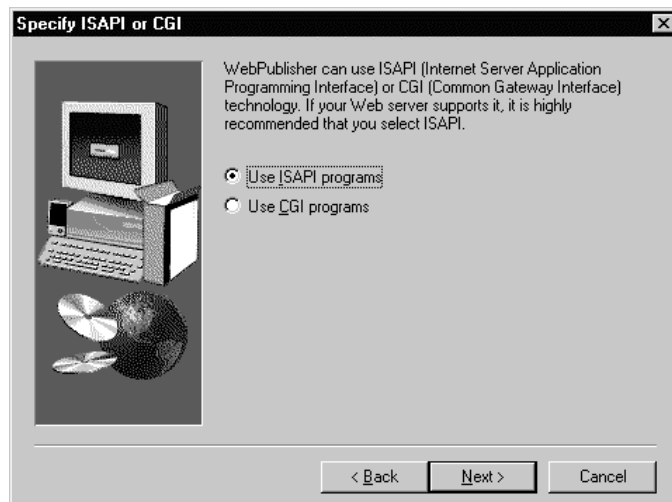
The URL you specify does not have to exist and will not be created automatically. It is simply a text string that *WebPublisher* will insert in front of textbase image file names. For example, if you specify /TBIMAGES, *WebPublisher* will construct URLs such as `http://hostname/tbimages/car.jpg`.

Note: Later, you will map the *WebPublisher* installation directory (for example, C:\PROGRAM FILES\INMAGIC\WEBPUB) to a virtual directory called /DBTW-WPD. If you plan to keep images in a subdirectory under the *WebPublisher* installation directory, use the virtual directory name in the URL. For example, type /DBTW-WPD/TBIMAGES/.

The URL you specify will be written to an initialization file called DBTW-PUB.INI. Later, you can edit that file to add or change locations. For more information, see “Textbase Record Images” on page 60.

11. Click **Next**.
12. In the Specify ISAPI or CGI dialog box, specify whether to use ISAPI or CGI programs, then click **Next**. If your Web server software supports it, we strongly recommend ISAPI. (IIS supports ISAPI.)

Note: Both programs will be installed. If you need to switch later, run ISAPI.BAT in the \JSCRIPT subdirectory to switch to using ISAPI, or run CGI.BAT to switch to using CGI.



13. View the READMEWP.HTM file and/or register online. We recommend doing both. Click **Finish**. *WebPublisher* Setup is now complete.



14. Map the installation directory (for example, C : \PROGRAM FILES\INMAGIC\WEBPUB) to a virtual directory called /DBTW-WPD, as explained in “Creating a Virtual Directory” on page 8.

Permissions

For a query to be sent over the Web to search a DB/TextWorks textbase with *WebPublisher*, you must have the proper permissions set for both your installation directory and virtual directory.

Note: The permissions needed for the virtual directory depend on the type of server software you are using (such as IIS, Netscape Enterprise, or O'Reilly WebSite). See “Creating a Virtual Directory” on page 8 for specific settings for various software types.

The Internet user account must have the necessary access rights to the *WebPublisher* directory tree shown on page 3. For important information and examples, see the *WebPublisher* READMEWP.HTM file. See your HTTP server documentation for information about the Internet user account. See your Windows documentation for information about using Windows NT Explorer to set permissions.

You may find it easiest to assign the following permissions for the Internet user account:

- Read-Execute to the full *WebPublisher* installation directory tree.
- Write access to the QSETS subdirectory (must be set after permissions for installation directory tree).
- Read-Write access to DBTW.PUB.INI in the Windows directory on the HTTP server. (Write access is preferred, but not required.)
- Read access to INMAGIC.INI and DBTEXT.INI in the Windows directory on the HTTP server (optional).
- Read access to the textbase directories specified during installation or by editing DBTW.PUB.INI.
- Read access to the directories that correspond to the virtual directories specified for images and logos during installation or by editing DBTW.PUB.INI.

Creating a Virtual Directory

A virtual directory called /DBTW-WPD is required for *WebPublisher* to function properly. This section briefly explains how to set up this virtual directory for various HTTP servers. For more information, or if your HTTP server is not listed, see your HTTP server documentation and/or the server's online help.

Mapping the *WebPublisher* installation directory (for example, C:\PROGRAM FILES\INMAGIC\WEBPUB) to the alias /DBTW-WPD enables *WebPublisher* to construct URLs for accessing information in the installation directory structure, and makes the two "locations" equivalent. For example:

C:\PROGRAM FILES\INMAGIC\WEBPUB\IMAGES\INMAGIC.GIF

is equivalent to the following address (where **hostname** is your domain name or IP address):

HTTP://HOSTNAME/DBTW-WPD/IMAGES/INMAGIC.GIF

Microsoft Internet Information Services 5.0 (Windows 2000)

1. Start the Internet Service Manager. For example, using the right mouse button, right-click the **My Computer** icon on your desktop. Select **Manage** from the menu that appears. Expand **Services and Applications**, then expand **Internet Information Services**.
2. Right-click **Default Web Site** and choose **New>Virtual Directory** from the menu.
3. Type dbtw-wpd in the **Alias** box to specify the alias to use to gain access to the virtual directory. Click **Next**.
4. Type the *WebPublisher* installation directory (for example, C:\PROGRAM FILES\INMAGIC\WEBPUB) in the dialog box that appears, and click **Next**. This is the physical path to the directory containing the content you plan to publish.
5. Select the **Read**, **Run Scripts**, and **Execute** check boxes on the dialog box that appears. Click **Next**, then click **Finish**. Note that these are the permissions for the virtual directory and are a key factor in *WebPublisher* working with your server.

Microsoft Internet Information Server 4.0

1. Start the Internet Service Manager. For example, choose **Programs>Microsoft NT 4.0 Option Pack>Microsoft Internet Information Server>Internet Service Manager**. (Note that if you are using a workstation, you will see **Microsoft Personal Web Server** instead of **Microsoft Internet Information Server**.)
2. Right-click **Default Web Site** and choose **New>Virtual Directory** from the menu.
3. In the **Alias to be used to access this virtual directory** box, type dbtw-wpd and click **Next**.
4. Type the *WebPublisher* installation directory (for example, C:\PROGRAM FILES\INMAGIC\WEBPUB) in the dialog box that appears, and click **Next**. This is the physical path to the directory containing the content you plan to publish.
5. Select the **Allow Read Access**, **Allow Script Access**, and **Allow Execute Access** check boxes in the dialog box that appears, and click **Finish**. Note that these are the permissions for the virtual directory and are a key factor in *WebPublisher* working with your server.

Microsoft Internet Information Server 4.0 - HTML

The HTML interface is an alternative to the dialog box interface described above.

1. Start the Internet Service Manager (HTML). For example, choose **Programs>Microsoft NT 4.0 Option Pack>Microsoft Internet Information Server>Internet Service Manager (HTML)**. You might have to start the Administration Web Server using the Internet Server Manager (non-HTML) first.
2. Select **Default Web Site** and click the **New** link at the top left of the screen.
3. Type `dbtw-wpd` in the pop-up dialog box that appears and click **OK**.
4. Select **dbtw-wpd** in the list and click the **Properties** link on the left side of the page.
5. In the Application Settings group, select **Execute (including Script)** from the droplist. Note that doing this sets permissions for the virtual directory and is a key factor in *WebPublisher* working with your server.
6. Click the **Create** button, then close the window.

Microsoft Internet Information Server 3.0

Note: The “lite” version of Microsoft Internet Information Server is called “Microsoft Peer Web Services.” Follow the procedure described below, but substitute “Microsoft Peer Web Services (Common)” in the first step.

1. Start the Internet Services Manager. For example, choose **Programs>Microsoft Internet Information Server (Common)>Internet Service Manager**.
2. Select the WWW service and start it if it is not already running.
3. Choose **Properties>Service Properties**.
4. Choose the Directories tab and click **Add**.
5. Type the *WebPublisher* installation directory (for example, `C:\PROGRAM FILES\INMAGIC\WEBPUB`) in the **Directory** box. This is the physical path to the directory containing the content you plan to publish.
6. Type `/dbtw-wpd` in the **Virtual Directory Alias** box.
7. Select the **Read** and **Execute** check boxes in the Access section. Note that these are the permissions for the virtual directory and are a key factor in *WebPublisher* working with your server.
8. Click **Apply**, then **OK**.

Netscape Enterprise Server 3.5.1

1. Start the Netscape Server Administration program. For example, choose **Programs>Netscape SuiteSpot>Administration**.
2. Type the Netscape Administration username and password when prompted.
3. Click the button for your HTTP server, under the Servers supporting general administration section.
4. Click the **Content Management** button at the top of the page.
5. Select **Additional Document Directories** from the list on the left.
6. Type dbtw-wpd in the **URL prefix** box. Type the *WebPublisher* installation directory (for example, C:\PROGRAM FILES\INMAGIC\WEBPUB) in the **Map To Directory** box. This is the physical path to the directory containing the content you plan to publish.
7. Click **OK**, then **Save** and **Apply**. Click **OK** on the message that confirms your changes have been made.
8. Click the **Programs** button at the top of the page.
9. Select **CGI Directory** from the list on the left.
10. Type dbtw-wpd/exec in the **URL prefix** box. Type the EXEC subdirectory under the *WebPublisher* installation directory (for example, C:\PROGRAM FILES\INMAGIC\WEBPUB\EXEC\) in the **CGI directory** box.
11. Click **OK**, then **Save** and **Apply**. Click **OK** on the message that confirms your changes have been made.

Note: Netscape Enterprise Server does not support ISAPI; you must use CGI. To switch to CGI, run CGI.BAT in the \JSCRIPT subdirectory. (If you need to switch back later for some reason, run ISAPI.BAT in the \JSCRIPT directory.)

O'Reilly WebSite

This information applies to versions 1.1f through 2.3. The user interface for later versions of O'Reilly WebSite may be different.

1. Start the O'Reilly WebSite Web Server. For example, double-click **Services** in the Windows Control Panel, select **Web Server** from the list, and click **Start**.
2. Right-click the O'Reilly WebSite taskbar icon and choose **Website server properties**.
3. Select the Mapping tab.
4. Select **Documents** in the List Selector, type /dbtw-wpd in the **Document URL Path** box, type the *WebPublisher* installation directory (for example, C:\PROGRAM FILES\INMAGIC\WEBPUB) in the **Directory** box. Click **Add**, then click **Apply**. This is the physical path to the directory containing the content you plan to publish.
5. Select **Standard CGI** in the List Selector, type /dbtw-wpd/exec in the **Standard CGI URL Path** box, and type the EXEC subdirectory under the *WebPublisher* installation directory (for example, C:\PROGRAM FILES\INMAGIC\WEBPUB\EXEC\) in the **Directory** box. Click **Add**, click **Apply**, then **OK**.

Note: O'Reilly WebSite does not support ISAPI; you must use CGI. To switch to CGI, run CGI.BAT in the \SCRIPT subdirectory. (If you need to switch back later for some reason, run ISAPI.BAT in the \SCRIPT directory.)

Chapter 2: Fundamentals

DB/Text *WebPublisher* provides a better way to publish, access, and maintain information on the Internet and on intranets. Use it to manage Web site information within the framework of a textbase, instead of static HTML documents. It is powerful but easy to use. Clients who access your Web site can search for information quickly and easily and display the results in a variety of ways.

DB/Text *WebPublisher* combines the power and precision of a database system with the speed and flexibility of a text retrieval product, giving you the best of both worlds.

Documentation and Other Resources

This manual assumes that you have a basic understanding of Internet issues and terminology, including HTML, URL, and the World Wide Web. It also assumes that you are familiar with DB/TextWorks and that you know how to create textbases, forms, query screens, and menu screens. For information about DB/TextWorks, see the *Inmagic DB/TextWorks User's Manual* and the DB/TextWorks online help.

In addition to this manual, you can use the following resources to learn about DB/Text *WebPublisher*:

- **Cars textbase.** A sample textbase you can use with the tutorials in Chapter 6.
- **Online help.** Help for DB/Text *WebPublisher* is integrated in the DB/TextWorks online help, to explain the many DB/TextWorks features that support *WebPublisher*. To access the help, start DB/TextWorks, choose **Help>Help Topics** or press **F1**, and search for topics on DB/Text *WebPublisher*.
- **READMEWP.HTM.** An HTML document that contains important information for textbase designers and Webmasters. It is located in the *WebPublisher* installation directory. It also details recently implemented *WebPublisher* features.
- **Knowledgebase and Web site.** Start DB/TextWorks and choose **Help>Inmagic on the Web>Knowledgebase** to go to the Inmagic Product Support knowledgebase on the Web, where you can search for solutions to common problems. Visit the Inmagic Web site (www.inmagic.com) by choosing **Help>Inmagic on the Web>Home Page** or **Help>Inmagic on the Web>Support Page**.
- **HTML help pages for clients.** HTML pages that explain query methods and error messages. For more information, see “How Clients Get Help” on page 65.

Note: If you have a Web browser, you can click the URL or link to go to the Inmagic Web site.

Terminology

This manual uses the following terms. For other definitions, see the Glossary on page 89.

- **Webmaster.** The server administrator.
- **Textbase designer or textbase administrator.** The person who uses DB/TextWorks to prepare textbases for Web access.
- **Client or user.** The person who searches the textbase by means of a Web browser.
- **Search screen, query screen, HTML search page.** An HTML page created with DB/TextWorks to allow clients to search a textbase.

Learning about DB/Text *WebPublisher*

Using DB/TextWorks and DB/Text *WebPublisher*, you can make information available to a wide audience on the World Wide Web or a corporate intranet.

What is DB/TextWorks?

DB/TextWorks is a textbase system—a type of database software that enables you to build networked and standalone textbases to manage text, numbers, dates, and electronic images. DB/TextWorks combines traditional database power, including the ability to manipulate data and perform arithmetic calculations, with the ability to handle large amounts of text. DB/TextWorks is referred to as “buildware” because you use it to build the textbases that you publish on the Web.

What is DB/Text *WebPublisher*?

DB/Text *WebPublisher* is a server version of DB/TextWorks that enables you to publish textbases on the Internet or an intranet, so Web users can search them using any standard Web browser. DB/Text *WebPublisher* consists of program files that accept queries submitted over the Web and return search results as dynamically formatted HTML reports. For the Webmaster, there is no user interface, so you will not see any icon on your desktop or on your Start menu.

Clients point their Web browser to a page on your Web site, where they can search for information by doing a query, instead of just clicking hypertext links.

DB/Text *WebPublisher* is ideal for publishing text-intensive information that is likely to change on a regular basis, including:

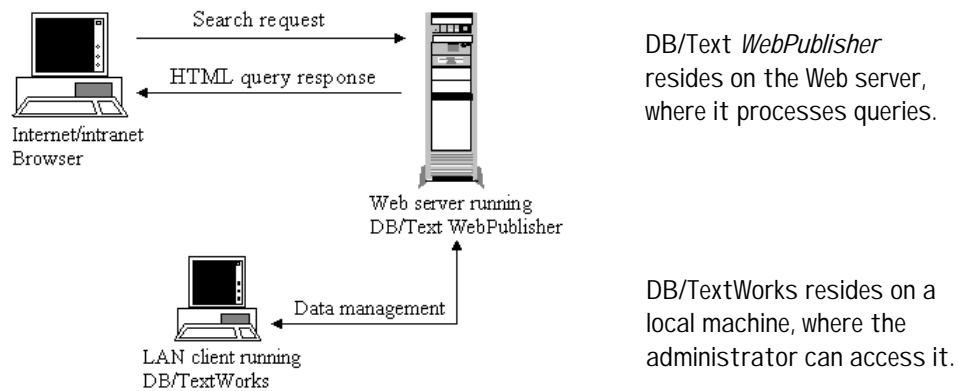
- Catalogs and other information products on an intranet or the Web
- Corporate intranet publishing, such as Human Resource policies and competitive intelligence documents
- Customer or prospect services, such as Help Desk and product catalogs

How Do They Work Together?

DB/Text *WebPublisher* combines the power of a Database Management System (DBMS) with the speed and flexibility of a text-retrieval system, to provide a superior method of Web site maintenance, access, and presentation.

Instead of having to produce separate, static HTML pages for your Web site, you can place information in a DB/TextWorks textbase and put that textbase on your server. Now anyone who goes to your Web site can search your textbases to find the information they need.

DB/Text *WebPublisher* can run in an Internet or intranet environment. The following illustration shows a typical LAN/WAN topology.



The following components are involved:

- **Browser Software.** The only client requirement is a Web browser, which clients point to a search screen to submit queries and display results.
- **HTTP Server Software.** Passes information between the browser and *WebPublisher*.
- **DB/Text *WebPublisher*.** Processes client queries and returns dynamically formatted results. There is no Windows user interface.
- **DB/TextWorks.** The buildware, a database application that the textbase designer uses to create and maintain textbases, forms, and HTML search screens.

What are the Key Features?

By integrating database management with Web publishing, DB/Text *WebPublisher* gives you a powerful, flexible method of publishing on the Web.

Administrators get a better way to manage information:

- **Robust data management capabilities for the Webmaster.** When information changes, you can use the batch modification, import, and delete functions to make edits quickly and easily in DB/TextWorks. Compare this to manual editing of separate HTML pages.
- **No knowledge of HTML required.** Use WYSIWYG tools to design search screens and reports. Conversion to HTML is done for you. Advanced users can add HTML to enhance their pages.
- **Open client/server environment.** DB/Text *WebPublisher* is a true 32-bit Windows application. It uses CGI or ISAPI on any PC running HTTP server software under a Win32 operating system. Clients do not need any special software, other than a Web browser. For more information, see “Client Requirements” on page 65.

Clients get an easier way to find and display information:

- **View actual data for which they can search.** Clients can click a button to see actual data from the textbase. They can paste words to search for, to find exactly what they want.
- **Fast, precise searching.** Searching is fast and precise, even through thousands of records.
- **Flexible search methods.** Clients can search multiple fields or run pre-defined searches, just by clicking a link. They are not limited to a single, broad keyword search that can turn up hundreds of unwanted hits. Instead, they can search multiple fields at the same time, achieving a high accuracy rate. Rather than searching for liberal arts colleges and getting 700 hits, they can search for co-ed liberal arts colleges in New England that have less than 2,500 students.
- **Search more than one textbase at a time.** Clients can search multiple textbases with a single query screen. For example, a client can search separate textbases containing books, periodicals, and research papers at the same time for a particular author. The results are formatted in a single, dynamically generated HTML report.
- **Dynamic display.** Content (records in the textbase) and format (layouts that you design) are controlled separately, so clients can see multiple views of the same data. Compare this to viewing a static HTML page, where the content and format are one and the same.

How Do I Publish a Textbase?

Publishing a textbase involves a textbase designer and a Webmaster. Depending on how your organization is structured, they may be the same person.

The designer uses DB/TextWorks to create a textbase, an HTML search page, and some forms for displaying search results. The designer passes the files to the Webmaster, who mounts them on the server.

The Webmaster sets up a link that points to the HTML search page. The Webmaster may also make changes to several initialization (INI) files, at the designer's request, to affect how *WebPublisher* operates.

To publish a textbase on the Web

1. The textbase designer uses DB/TextWorks to perform these tasks:
 - a. **Create a textbase and populate it with records.** This is the “text database” that *WebPublisher* will search.
 - b. **Create forms to display records found by a search.** Forms determine how records appear in the Web browser.
 - c. **Create a query screen and export it to HTML.** This is the HTML search screen that clients will use to submit queries.
 - d. **[Optional] Create a menu screen and export it to HTML.** This is an HTML page that contains links. Clicking a link runs a pre-defined query.
2. The textbase designer coordinates the following issues with the Webmaster. For more information, see Chapter 5, “For the Webmaster,” on page 67.
 - **Textbase file locations.** DB/TextWorks and DB/Text *WebPublisher* need access to the textbase. The textbase can be in a shared location, or you can store the textbase locally and copy it up to the HTTP server whenever you make changes.
 - **HTML and image file locations.** Clients need access to the HTML search pages you create. Coordinate access to HTML pages and associated image files with the Webmaster.
 - **Initialization files.** Initialization files stored on the HTTP server control many *WebPublisher* settings. From time to time, you may need to edit these files.
 - **Intranets.** If you are publishing on an intranet, you can provide access using HTTP or network file addressing protocols. For more information, see “Launching Applications on an Intranet” on page 61.
3. Now the textbase is ready to be searched by clients. Be sure to test it thoroughly before making it available on the World Wide Web or your corporate intranet.

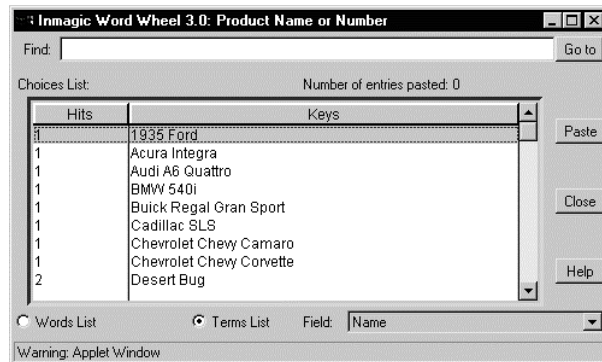
How Do Clients Use *WebPublisher*?

Anyone with a Web browser can point to an HTML search screen and submit a query.

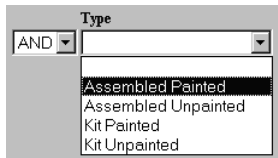
To make searching even easier, the designer may have included Word Wheel buttons and choices droplists, as shown below. Clients can select items to search for from a Word Wheel or choices droplist. These query methods eliminate guesswork, making searching easier and more intuitive.



A Word Wheel is a dialog box that shows indexed information from the textbase.

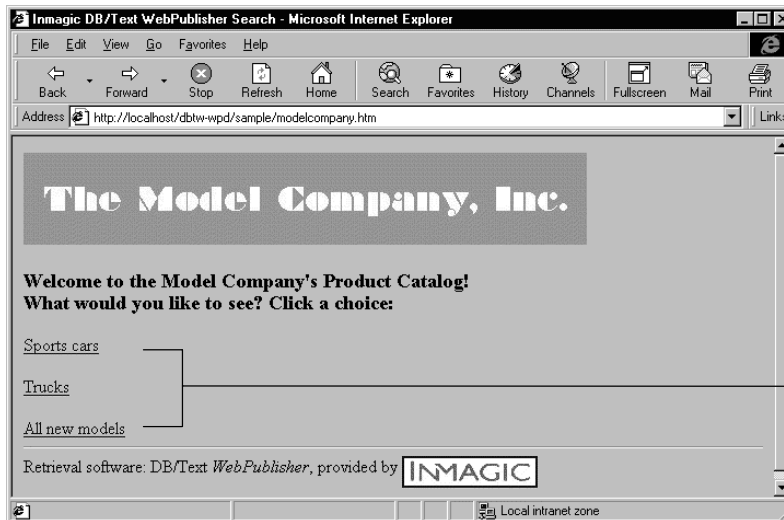


A choices droplist lets clients select items from a list and is useful for fields that contain just a few entries.



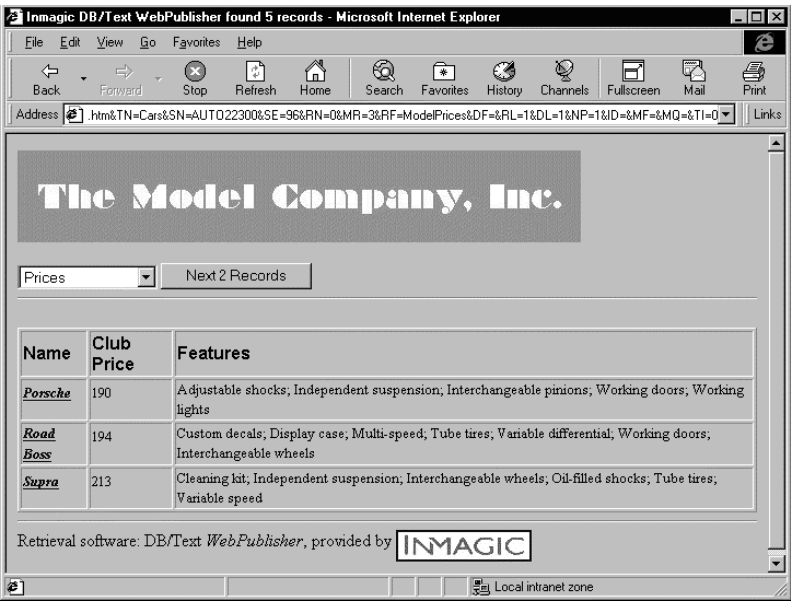
For more detailed information about choices droplists and Word Wheels, see “Choices Droplist” on page 47 and “Word Wheel” on page 48.

The textbase designer may also have supplied a menu screen, which is an HTML page that includes one or more pre-defined queries. To the client, the pre-defined queries look like regular hypertext links. Clicking a link submits a query directly to *WebPublisher*, which processes the query and returns results.



On a menu screen, clients click links to run pre-defined queries.

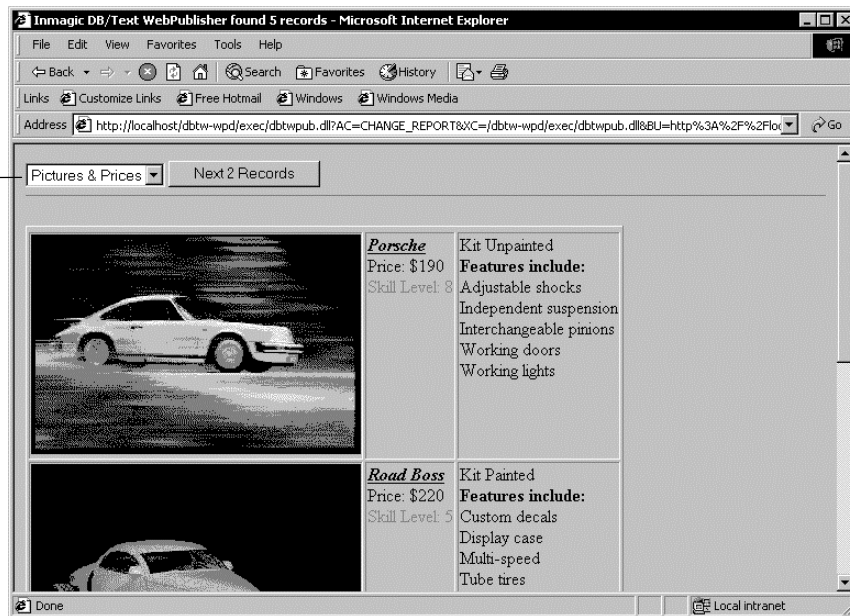
When a client submits a query—either by clicking a link on a menu screen or clicking the **Execute Query** button on a Web search screen—the search criteria are passed to DB/Text *WebPublisher*, which performs the query, formats the results (or an initial subset of results), and returns the information to the Web browser in HTML format.



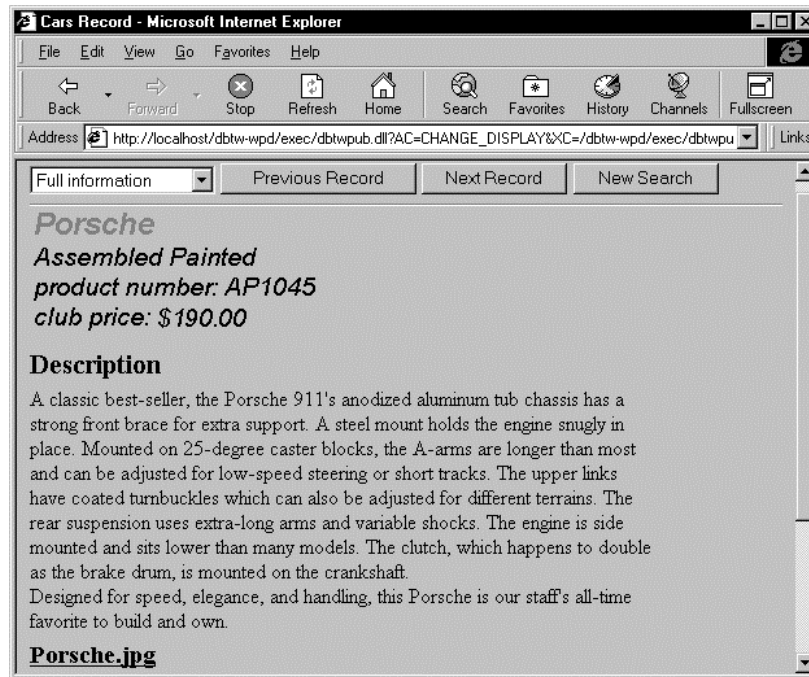
A typical report shows a summary of records found.

The client can use the standard Web browser controls to look at the records, return to the home page, print, and so forth. DB/Text *WebPublisher* provides other controls, such as navigation buttons on the report. Record appearance is controlled by forms that are designed with DB/TextWorks. Clients can change record appearance dynamically by selecting different forms from a droplist in the browser. The following illustration shows the same records from the previous illustration, but after the client selected a different form.

Select a
different
form to see
a different
view of the
records.



Forms that display multiple records (such as the forms in the previous two illustrations) often include a hypertext link to a single-record display. Clients can click the link to see a detailed view of one record at a time. This is called expanding a record. For example, if you click the **Porsche** link on the report shown in the previous illustration, you see a single-record display.



An expanded view shows one record at a time.

After expanding a record, the client can perform some or all of these tasks:

- Display the next or previous record using the **Next Record** or **Previous Record** buttons.
- Select a different form from the form droplist to change the appearance of the record.
- Return to the report by clicking the browser's **Back** button, or by closing the browser window if the expand record opens in a separate browser window. (For more information about the parameter needed for an expanded record to open in a separate browser window, see page 30.)
- Print records, using the currently selected form to format the output. The Web browser software may also have an effect on output (for example, by numbering pages).
- Return to the query or menu screen using the **New Search** button. Note that you must add a parameter to the DBTW PUB.INI file for this button to appear, and the expanded record must open in the same browser window as the report. This feature is also available in the Report window. For more information, see the READMEWP.HTM file.

Chapter 3: Using the DB/TextWorks Buildware

This chapter explains how to create textbases, forms, and search screens for Web use.

Textbases

Use DB/TextWorks to create the textbases that clients will search. A textbase consists of multiple files with the same name but different extensions (for example, *CARS.**). You can use the textbases locally, publish them on the Web, or both.

Creating a Textbase

Creating a textbase is easy to do. To get started quickly, follow the instructions below. For more information, see the *Inmagic DB/TextWorks User's Manual* and the online help.

1. Start DB/TextWorks.
2. Choose **File>New Textbase** to open the Specify New Textbase dialog box. Specify a file name and location, then click **Save**.
3. On the Create Textbase Structure dialog box, select the **Define New Textbase** option button and click **OK**.
4. Use the Edit Fields dialog box to specify the fields that will make up your textbase. For example, type **Title** in the **Field Name** box and click **Add**. Do this for each field you want to add to the textbase.
5. Select a field from the Field Name list, then select a field type for it from the **Field Type** droplist on the Type and Indexing tab. Also specify indexing information and Special Filing Options, if applicable. Click the **Change** button.

Note: You can use the other tab(s) on this dialog box to specify other attributes for your fields. Click the **Help** button on the Edit Fields dialog box for more information about this dialog box.

6. Click **Close**.
7. Click **OK** on the Edit Textbase Structure dialog box.
8. A message indicating the textbase is empty appears. Click **OK**.

9. The Edit: New Record window opens, so you can begin adding records to the textbase. Type information in the boxes, then choose **Records>Save Record** to save the record.
10. To add another record, choose **Records>New Record**.

Maintaining Information in a Textbase

You can use the robust data management capabilities of DB/TextWorks to update the information being published on the Web. Just open the textbase in DB/TextWorks and use one of the methods listed below. For more information about these features, see the *Inmagic DB/TextWorks User's Manual* and the online help.

- **Batch Modify (Records menu).** Add an entry, replace or delete entries, and change or delete text within entries, in a group of records.
- **Batch Delete (Records menu).** Delete a group of records.
- **Import (File menu).** Import Inmagic tagged, delimited ASCII, and XML files containing multiple records. During an import, you can add new records, replace existing records with modified records, add to or replace fields in existing records, or delete existing records.
- **Import Document (File menu).** Import the complete text of one or more documents, such as letters and EMail messages. Add each imported document as a new record, or add or replace text in existing records. To import word-processed documents in many native formats, including Microsoft Word and WordPerfect, you can use an add-on product called DB/Text® *Import Filter*.
- **Edit Record (Records menu).** Search for one or more records, then edit each record individually.

Textbase Locations

Both DB/TextWorks and *WebPublisher* need access to the textbase. The textbase can be in a shared location or you can store the textbase locally and copy it up to the HTTP server whenever you make changes. For important information, see Chapter 5, "For the Webmaster," on page 67.

Other Considerations

When creating textbases that you plan to publish on the Web, consider the following issues:

- Indexing
- Passwords
- Textbase default forms
- Sort order

Each feature is summarized below. For more detailed information, see the *Inmagic DB/TextWorks User's Manual* or the online help.

Indexing

You cannot search a field unless it is indexed. Use DB/TextWorks to add a Word and/or Term index to every field you will allow clients to search. It is good practice to Word index every searchable text field. If a field is Term indexed only, word and phrase searches may not retrieve the desired records.

Passwords

You can use passwords to hide and protect information. For menu screens, the textbase must have a Silent password or clients will not be able to search. For query screens, you can either use a Silent password or add a password box to each query screen that you will publish on the Web. If you use an Access Control field type in your textbase, passwords can also be used to restrict access to individual records, such as catalog records that are on order.

To add a password box

1. Choose **Maintain>Edit Textbase Structure** and click the **Passwords** button to open the Textbase Passwords dialog box.
2. Specify a Master password and at least one Field Access password and/or a Silent password by using the appropriate tabs on the Textbase Passwords dialog box.

Note: You cannot password-protect droplists or Word Wheels on a query screen.

3. If you want to restrict access to classes of records within the textbase (and your textbase has an Access Control field), click the **Record Level Security Settings** button and specify the access you want allowed for each Field Access and Silent password, as applicable. For more information about restricting access to records see the online help.
4. Open a query screen in the Query Screen Designer and choose **Tools>Screen Properties** to open the Screen Properties dialog box.
5. On the HTML tab, select the **Add an 'Enter password:' box** check box and click **OK**.

Note: If a textbase does not have passwords, you can design forms that omit the information you want to hide. However, if no initial forms or textbase default forms are specified, a form is generated that shows all non-hidden fields.

Textbase Default Forms

It is a good idea to select textbase default forms for reports and single-record display. Open a textbase in DB/TextWorks, choose **Maintain>Change Textbase Defaults>Forms**, and use the **Record Display** and **Report Window** buttons. (The Record Edit and Report Printing defaults are for desktop use only.)

The default forms that you select will be used if you do not specify initial forms when designing a query screen or menu screen for export to HTML. If no initial or default forms are specified, records will be displayed using an automatically generated Basic form, which shows all non-hidden fields with minimal formatting. The forms that you select as defaults or initial forms must be saved as **WebPublisher Reports**. This is specified when you save the form for the first time, or when you choose **Form Operations>Save Form As** in the Form Designer.

Sort Order

Sort order is determined by the textbase default sort order (choose **Maintain>Edit Textbase Structure>Sort Order**) or by the compulsory sort specified for the form. To ensure sensible displays in the Web browser, use DB/TextWorks to specify a logical default sort order, such as alphabetically by title for a textbase of documents, and/or assign a compulsory sort to report forms that will be used on the Web.

Forms

Forms are layouts that determine how records appear in the Web browser after a search. Clients can select forms from a droplist in the Web browser after a search to see different views of the records retrieved.

You design a form for Web use the same way you design forms for the desktop, with the exceptions and additions described in this chapter. The *Inmagic DB/TextWorks User's Manual* and online help are your best sources for information about designing forms. For more information about designing forms for the Web, look up "Tips" in the DB/TextWorks online help.

For each textbase that you will publish on the Web, create at least one report form and at least one display form.

- **Report forms.** Summarize multiple records found by a search. Tabular forms are good for this purpose. Report forms often include a hypertext link to a display form.
- **Display forms.** Used when a user clicks on an expand record link. Shows detailed information about one record at a time.

Search results are displayed in the Web browser using a report form, such as the one illustrated below, which shows information from the *Title*, *Author*, and *Summary* fields.

Title	Author	Summary
ABCs of Finance	Ingrid Spy	A beginner's guide to finance.
Financial Trends	John Dough	Analysis of financial trends.
Managing Your Moolah	Mary Jones	A perennial favorite.

A report form designed for Web use often includes an expand record link (a link to an expanded display form). The form in the previous illustration uses the *Title* field as the expand record link. If you click on a title, such as **ABCs of Finance**, you will see more information about that book.

ABCs of Finance, by Ingrid Spy

Published in May 1996 by Small Town Press.

475 pages

Hardcover only

This comprehensive guide to money management is ideal for the new investor. Written by the guru of high finance and filled with practical examples, this tome is a mighty addition to the home library.

As far as the client is concerned, he or she has jumped to more information about the book. What has actually happened is that the single record has been displayed using a display form.

Tip! You can view the expanded record in a second browser window by adding a parameter to the DBTWPub.INI file. This lets you close the window after viewing single records instead of having to press the Back button on your toolbar multiple times to get back to the multiple record report. For more information, see page 30 or the READMEWP.HTM file.

Designing Report Forms for Web Use

For each textbase that you will publish on the Web, use the Form Designer to create at least one report form that shows a summary of records found. Using a tabular form is a typical and effective way to show a summary of records. For more information, see “Designing Tabular Report Forms for Web Use” on page 29.

To design a report form for the Web

1. Start DB/TextWorks and open a textbase.
2. Open the Form Designer by choosing **Display>Design Form**, then use the Open Form dialog box to open an existing form or create a new one.

3. Add form, text, and/or picture boxes to the form, as applicable. Each box has a different type of content, which is seen when information is displayed on the Web. For example, if you put the *Name* field in a form box, you will see names in the box when the form is used. A report form should show just a few of the most important fields.

Note: While you can add script input boxes and script buttons to query screens, they are ignored when the query screen is exported to HTML because they are for desktop use only.

- To add a form or text box, choose **Edit>Add>Form Box** or **Edit>Add>Text Box**. Use the Box Properties dialog box to add content and/or text to the box, as well as specify other attributes, as appropriate.
 - To add a picture box, choose **Edit>Add>Picture Box**. Use the Picture Box Properties dialog box to designate the image file or field you want to include in the form. Be sure you place the images in a location where *WebPublisher* will look for them. For more information about displaying images, see “Choosing the Best Image Display Method” on page 59 and “Image File Locations for Forms” on page 70.
4. For form and text boxes, use the HTML tab on the Box Properties dialog box to specify properties for the box when used on the Web. (The options specified here are not applicable to desktop use.) Use the **Treat content item as** droplist for form boxes and the **Treat text as** droplist for text boxes to specify important Web-specific properties. For example, use these droplists to indicate which text to use as the link to the single-record display, as explained on page 32. For more information, see “Adding HTML Properties to a Box” on page 34.
 5. Choose **Tools>Form Properties** to specify properties that apply to the form as a whole. For example, you may want to clear the **Highlight search items** check box on the General tab, as highlighting can be distracting in a report. Use the Logos and HTML tabs to select important Web-specific properties, as explained on page 33.
 6. Use the Report Options menu to specify options for multiple-record display. (Note that the Margin Area only applies to forms used for printing on the desktop.)

7. Choose **Form Operations>Save Form As**; then name the form and select these options:
 - **Textbase File (Public)**. This saves the form inside the textbase. This option button **must** be selected for a form to be used on the Web.
 - **Report Window**. This makes the form available for reports.
 - **WebPublisher Reports**. Only forms that have this box selected will be available to Web clients. You can also select **WebPublisher Only**, to omit a form from desktop picklists, if the form is not suitable for desktop use.

Important! There are many other features that you can use to design a report form. For more information, see the *Inmagic DB/TextWorks User's Manual* and the online help.

Designing Tabular Report Forms for Web Use

Design a tabular report form if you want to present information in rows and columns. Each row in the table represents a record. Each column represents a field. Each cell contains one or more fields or other content items (for example, variables, calculations).

Tabular forms are a useful way to display a summary of records found by a search, showing just a little information from each record. The client can expand each record to see more information. The tabular form shown below uses the *Title* field as the expand record link.

Title	Journal
<u>Brilliant Corners.</u>	Village Voice
<u>The Modern Jazz Quartet Reform.</u>	Downbeat
<u>All That Jazz.</u>	Black Enterprise
<u>Jazz Goes to College.</u>	Ebony
<u>Ad Agencies Jazz Up Jingles by Playing 1960s Nostalgia.</u>	Wall Street Journal
<u>Will Your School Be Next?</u>	Dance Magazine
<u>Editor's Note: I Need My Plastic Wrap.</u>	American Demographics

To create a tabular form

1. Open a textbase in DB/TextWorks.
2. Choose **Display>Design Form** to open the Open Form dialog box.
3. Select **New Tabular Form** from the Start With list and click **OK**.
4. On the Choose Initial Tabular Form Fields dialog box, specify which fields you want to include in the table. Select a field from Available Fields list and click the > button to move it to the Initial Fields list. Do this for each field you want to add to the table. Click **OK**.

5. Select a box and choose **Tools>Box Properties** to open the Form Box Properties dialog box. Specify content, position, label, formatting, and HTML attributes, as applicable. Include an expand record link, so users can display one record at a time. See the next section, “Adding an Expand Record Link to a Report,” for information on how to do this.
6. Save the form.

Adding an Expand Record Link to a Report

When you create a report form, it is important to specify which text to use as the hypertext link to an expanded display, so users will have a way to see one record at a time. You can create a link using form box or text box content.

To create an expand record link using form box content

1. Open a report form in the Form Designer.
2. Select a form box whose content will be the hypertext link. As a general rule, choose a field that is present and different in every record, such as the title of a publication.
3. Choose **Tools>Box Properties** to open the Form Box Properties dialog box.
4. On the HTML tab, select a content item from the Contents list, then select **Expand record link** from the **Treat content item as** droplist.
5. Click **Apply**.

To specify which form to use for expanded displays, see “Specifying Initial Forms” on page 38.

The form shown below uses the *Title* field as the expand record link. Clicking on a title will show detailed information about one record at a time.

Title	Author	Summary
<u>ABCs of Finance</u>	Ingrid Spy	A beginner's guide to finance.
<u>Financial Trends</u>	John Dough	Analysis of financial trends.
<u>Managing Your Moolah</u>	Mary Jones	A perennial favorite.

When you click on an expand record link, the record opens in the existing window. You can optionally specify that it open in a new Web browser window. This makes it easy for the user to return to their report by closing the expanded record window. To do this, add the WebExpandInSeparate parameter to the [WebPublisher] section of the DBTWPUb.INI file:

```
[WebPublisher]
WebExpandInSeparate=1
```

To create an expand record link using text box content

- Follow the steps above, but select a text box and select **Expand record link** from the **Treat text as** droplist.

Note: When you add the text box to the form, type appropriate text in the **Text** box so users know why they should click on that link. For example, type **More** or **Full Record**.

The following form shows a text box with the word **More** used as an expand record link. Note that you can also use fixed text in a form box to achieve the **More** link illustrated below.

Title	Author	Summary	
ABCs of Finance	Ingrid Spy	A beginner's guide to finance.	More
Financial Trends	John Dough	Analysis of financial trends.	More
Managing Your Moolah	Mary Jones	A perennial favorite.	More

Sorting Records in Reports

Sort order is determined by the textbase default sort order (choose **Maintain>Edit Textbase Structure>Sort Order**) or by the compulsory sort specified for whichever form is in use. To specify a compulsory sort, choose **Report Options>Compulsory Sort** in the Form Designer.

To give the client some control over sorting, you can design reports for the Web that are identical except for different compulsory sorts. For example, provide reports called Sort by Title and Sort by Date. To learn more about compulsory sorts and the textbase default sort order, see the *Inmagic DB/TextWorks User's Manual* and the online help.

Note: The following compulsory sort options are ignored by *WebPublisher* because they can change the perceived number of records retrieved: **Exploded Sort**, **Interfile**, and **Omit Those Records** (click the **Primary Sort Field Options** button).

Limiting the Number of Records Displayed

You can limit the number of records displayed per page of a report. For example, you can specify that a maximum of 20 records will appear on a report page. This does not limit the number of records found; it just controls how many are displayed on a page. To see more records, or to return to a previous page, clients can click the next record and previous record buttons on the report page.

The query or menu screen that submits the query controls the number of records displayed and the placement of the next record and previous record buttons and form droplists. For more information about limiting the number of records displayed and placing the next record and previous record buttons, see “Adding HTML Properties to a Query Screen” on page 44 and “Adding HTML Properties to a Menu Screen” on page 57.

Designing Expanded Display Forms for Web Use

For each textbase that you will publish, create at least one expanded display form that shows detailed information about one record at a time.

To design a display form for the Web

1. Start DB/TextWorks and open a textbase.
2. Open the Form Designer by choosing **Display>Design Form**, then open an existing form or create a new one using the Open Form dialog box.
3. Add form, text, and/or picture boxes to the form, as applicable. Each box has a different type of content, which is seen when information is displayed on the Web. For example, if you put the *Name* field in a form box, you will see names in the box when the form is used. An expanded display form should show most of the information in a record.

Note: While you can add script input boxes and script buttons to forms, they are ignored in forms used on the Web because they are for desktop use only. This is because form scripts are ignored in *WebPublisher*.

- To add a form or text box, choose **Edit>Add>Form Box** or **Edit>Add>Text Box**. Use the Box Properties dialog box to add content and/or text to the box, as well as specify other attributes, as appropriate.
 - To add a picture box, choose **Edit>Add>Picture Box**. Use the Picture Box Properties dialog box to designate the image file or field you want to include in the form. Be sure you place the images in a location where *WebPublisher* will look for them. For more information about displaying images, see “Choosing the Best Image Display Method” on page 59 and “Image File Locations for Forms” on page 70.
-

Tip! To edit box content and appearance, select a box, then use the Box Properties dialog box (choose **Tools>Box Properties**). For example, for form boxes, select the Labels tab to add or remove the box label. Select the Position tab to re-arrange or resize boxes. You can also use the mouse to move and resize boxes. See the online help for more information.

4. For form and text boxes, use the HTML tab on the Box Properties dialog box to specify properties for the box when it is used on the Web. (The options specified here are not applicable to desktop use.) Use the **Treat content item as** droplist for form boxes or the **Treat text as** droplist for text boxes to specify important Web-specific properties. For more information, see “Adding HTML Properties to a Box on a Form” on page 34.
5. Choose **Tools>Form Properties** to specify properties that apply to the form as a whole. For example, add a background color by selecting the **Background color** check box on the General tab and clicking the **Select Color** button to open the Color dialog box. Use the Logos and HTML tabs to select important Web-specific properties, as explained on page 33.
6. Choose **Form Operations>Save Form As**, then name the form and select these options:
 - **Textbase File (Public)**. This saves the form inside the textbase. This option button **must** be selected for a form to be used on the Web.
 - **Record Display Window**. This makes the form available for single-record display.
 - **WebPublisher Reports**. Only forms that have this box selected will be available to Web clients. You can also select **WebPublisher Only**, to omit a form from desktop picklists, if the form is not suitable for desktop use.

HTML Properties for Forms

HTML properties can apply not only to forms as a whole but also to individual boxes within forms. When you design a form for the Web, you can select Web-specific options from the tabs listed below.

- **Tools>Form Properties>Logos**
- **Tools>Form Properties>HTML**
- **Tools>Box Properties>HTML**

HTML properties only affect forms used on the Web. They are ignored for forms used on the desktop. For more information, select the tab for which you want help on the Form Properties, Form Box Properties, and/or Text Box Properties dialog box, then click the **Help** button.

Adding Logos to a Form

Use the Logos tab on the Form Properties dialog box (choose **Tools>Form Properties>Logos**) to add images to the top and/or bottom of a form.

Type the image name in the **Leading Logo** and/or **Trailing Logo** box, or click the **Browse** button to navigate to it. For more information, see “Choosing the Best Image Display Method” on page 59 and “Logo and Background Image File Locations” on page 71.

Adding HTML Properties to a Form

Use the HTML tab on the Form Properties dialog box (choose **Tools>Form Properties>HTML**) to add the following HTML-related properties to a form:

- **Use record separators.** Select this check box to add and define a horizontal rule between records in a report.
- **Use background image.** Select this check box to specify an image file for the page background. For more information, see “Choosing the Best Image Display Method” on page 59 and “Logo and Background Image File Locations” on page 71.

Note: To specify a color for the form background, choose **Tools>Form Properties>General**.

- **Table settings.** Add a border around a table by specifying a value in the **Border width** box. Use zero (0) if you do not want to show a border. This option appears for tabular forms only.
- **Advanced Options.** Click this button to specify information you want included in the <HEAD> section of the HTML page. The information (<SCRIPT>, <STYLE>, or other HTML elements) is inserted after the information added by the software, such as the page title.

Adding HTML Properties to a Box on a Form

Use the HTML tab on the Box Properties dialog box (choose **Tools>Box Properties>HTML**) to specify whether and how the content of a form or text box will be interpreted as HTML.

Note: HTML box properties do not apply to picture boxes. Script input boxes and script buttons are ignored in *WebPublisher*.

For form boxes, select one of the following options from the **Treat content item as** droplist. For text boxes, select one of the following options from the **Treat text as** droplist. The type of box selected determines which droplist appears.

Note: The options for form and text boxes are the same, except for a few. The “See Also” search link is only applicable to form boxes, as are the Inline image and Image link options. Those options do not appear in the **Treat text as** droplist.

- **Do not alter.** Use this option for form and text boxes that contain information that you do not want to be interpreted as HTML by the client’s Web browser, such as titles and instructions. The text will appear in the browser as it appears in the form and/or text box. The Web browser will not attempt to interpret any characters as HTML tag delimiters.

- **Raw HTML.** Use this option to pass the box item through to the Web browser. When the form is used on the Web, the browser will interpret the text as HTML. For example:

```
<a href="http://www.myco.com/query.htm">Return to query screen</a>
```

appears on the report as a hypertext link:

Return to query screen

For important information about using Raw HTML, see “Cascading Style Sheets and Absolute Positioning in Forms” on page 37.

- **URL.** Use this option if you want DB/Text *WebPublisher* to add the necessary HTML tags to turn URLs (for example, `http://www.inmagic.com`) into hypertext links. If a form box contains a field with multiple URLs as repeating entries, selecting **URL** will apply to all of the URLs. A text box on a form can only contain a single URL.
- **Email link.** Use this option if you want *WebPublisher* to interpret the contents of a form or text box as a “mailto” link. For example,
joe@mydomain.com

will be treated as:

```
<a href="mailto:joe@mydomain.com">joe@mydomain.com</a>
```

If a form box contains a field with multiple EMail addresses as repeating entries, selecting **Email link** will apply to all of the addresses. (A text box on a form can only contain a single EMail address.)

- **HTML file reference.** Use this option to treat a content item or text as a file reference (file://). This option is typically used to launch an application on an intranet. For example:
\\server\volume\myfile.doc

is treated by *WebPublisher* as:

```
<a href="file://server/volume/myfile.doc">\\server\volume\myfile.doc</a>
```

To display alternate text on the screen, in place of the file reference, select the **Use alternate link text** check box. Type the alternate text (for example, `Click here`) in the **Text** box.

Users see the words `Click here`. If the field contains repeating entries, the same alternate link text will appear for each entry. For more information about using HTML file references to launch applications, see “Launching Applications on an Intranet” on page 61.

- **Expand record link.** Use this option to add a link from a multiple-record display to a single-record display. You can create expand record links for form and text boxes. For more information about how to create these links, see “Adding an Expand Record Link to a Report” on page 30.
- **“See Also” search link.** Use this option (for fields in form boxes only) to treat each entry in a field as a link that will search the textbase. For example, turn the *Author* field into a “See Also” link that, when clicked, finds all other books by the same author. If this option is used and a book has multiple authors, each author becomes a link to his/her other works.

- **Inline image.** Use this option (for form boxes only) to display images as part of the form. If you choose this option, check to be sure that your image dimensions are an appropriate size. You may have to use a graphics or conversion application to reduce the dimensions. If the selected field contains multiple entries, and images are displayed inline, Web users will be able to see all the images.

Note: To display only the first image, use a picture box. Picture boxes also let you specify the image size without modifying the original image file.

- **Image link.** Select this setting (for form boxes only) to specify that each image name in a field in the textbase record be treated as a link to that image. For more information, see “Choosing the Best Image Display Method” on page 59, “Textbase Record Images” on page 60, and “Image File Locations for Forms” on page 70.

For details about these HTML properties, see the DB/TextWorks online help.

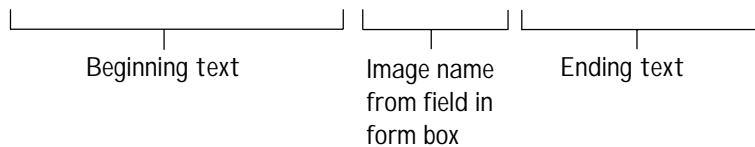
Adding Your Own HTML Tags

Those familiar with HTML (Hypertext Markup Language) code can optionally use HTML tags in text and form boxes to format their forms.

For example, you can use HTML as beginning, ending, and/or separator text on forms designed for use with DB/Text *WebPublisher*. Use HTML to wrap around the contents of a form box.

This means, for example, that you can create a hyperlink on your form that, when clicked, will display an image from a field containing an image file name. (The image will correspond to the record being displayed in the form.) To do this, type the following HTML code in the appropriate boxes on the Added Text subtab of the Format tab on the Form Box Properties dialog box. (Notice in the example below that the ending text begins with the closing quotation mark around the image name.)

```
<a href="http://images/kitten.jpg">Click here</a>
```



Note: Content items and text that you do not treat as **Raw HTML** are automatically formatted by *WebPublisher* (for example, by enclosing them in tags, and so forth). That means that advanced users who want to format all of the elements in their form with HTML must be sure that every content item and all text in a box is treated as **Raw HTML**. Doing this enables the browser to read the HTML code they include, rather than using the automatic *WebPublisher* formatting.

Cascading Style Sheets and Absolute Positioning in Forms

WebPublisher uses Cascading Style Sheets (CSS) and absolute positioning so the report and display forms you design in DB/TextWorks translate as well as possible to the Web.

Using CSS allows you to retain box size, including picture boxes, on the Web. Position is also retained, meaning you can have side-by-side boxes, and left labels.

Versions 4 or later of Internet Explorer and Netscape Navigator usually support Cascading Style Sheets. If a browser does not support CSS, *WebPublisher* detects it and will use simple HTML to display the form on the Web. If this happens, boxes will be left-justified, box labels will appear on top of boxes, forms will use browser default font faces, and so forth.

Forms that contain Raw HTML or inline images are incompatible with absolute positioning. Forms containing these elements will have overlapping text unless very carefully designed. Therefore, *WebPublisher* uses simple HTML for this type of form, unless you explicitly set WebCSOpt=2 in your DBTWPub.INI file (see below).

Note: DB/TextWorks tabular forms do not use absolute positioning, so this issue does not have an effect on them.

You can use picture boxes in place of inline images to retain CSS formatting. The options selected in the Image size group on the Picture Box Properties dialog box are used (choose **Tools>Box Properties>Position**). In order for the image sizing to work properly, the images must reside either in the textbase directory (for example, C:\CATALOG) or the directory specified in the Image field in the record (for example, C:\IMAGES\COYOTE.JPG).

To control whether the site uses CSS, modify the [WebPublisher] section of your DBTWPub.INI file to include the following:

```
[WebPublisher]
WebCSOpt=1
```

The switches set the following:

0 = do not use CSS (use simple HTML as described above)

1 = the default behavior (use CSS unless the form contains Raw HTML or inline images)

2 = use CSS regardless of whether forms contain Raw HTML or inline images

Note: The WebCSOpt= setting affects all forms and all textbases.

Specifying Initial Forms

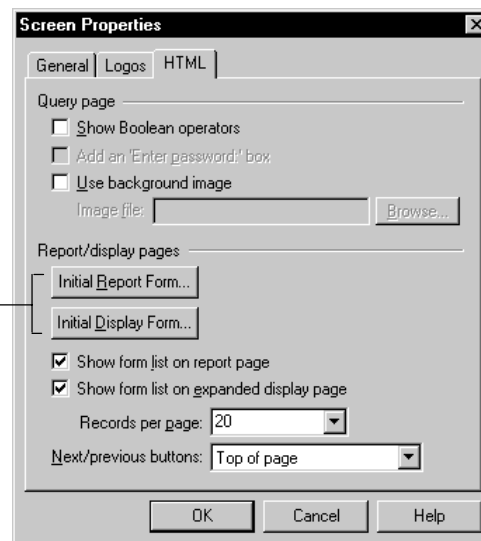
An initial form is the form that is first used in the browser. The initial report form is used when the client retrieves records. The initial display form is used when the client expands a record. Clients can change forms using a form droplist in the Web browser. If you do not specify initial forms, the textbase default forms will be used (if they were saved as *WebPublisher* reports), or a Basic form will be generated that shows all non-hidden fields with minimal formatting.

Note: The query or menu screen that submits the query determines which forms are used.

To specify initial forms for query screens

1. Choose **Search>Design Query Screen** and select the query screen that will submit queries.
2. Choose **Tools>Screen Properties** to open the Screen Properties dialog box.
3. On the HTML tab, click the **Initial Report Form** and/or **Initial Display Form** buttons and use the Select Report Form and/or Select Record Display Form dialog boxes to specify initial forms to use when queries are submitted using this page.

Use the **Initial Report Form** and **Initial Display Form** buttons to specify the forms used when queries are submitted over the Web.



4. Specify whether a form list appears on the report page and/or expanded display page by selecting or clearing the **Show form list on report page** and **Show form list on expanded display page** check boxes, as appropriate.
5. Specify the number of records you want displayed on one report page after a search, or select **unlimited** from the **Records per page** droplist. Limiting the number of records displayed on a page lets users view the records matching their search criteria in batches.

6. Use the **Next/previous buttons** droplist to specify where the form droplist and the next record and previous record buttons will appear. This setting affects all *WebPublisher* forms used when a query is submitted through this screen. It also controls where the optional **New Search** button will appear on the page (even if the next/previous record buttons and form droplist do not appear).
7. Click **OK**.

To specify initial forms for menu screens

1. If a textbase is open, close it, then choose **Menu Screens>Design** and open the menu screen that will submit queries.
2. Select a menu item that has a saved query specified as its Initial Action.
3. Choose **Tools>Box Properties** to open the corresponding Box Properties dialog box.
4. On the **Initial Elements** tab, click the **Specify Initial Elements** button to open the Specify Initial Elements dialog box.
5. Click the **Forms** button to open the Specify Initial Forms dialog box.
6. Use the **Report Window** and **Record Display** buttons to specify forms for those windows.

Note: Be sure to select forms that were saved as **WebPublisher Reports** (in the Save Form As dialog box). Forms must be saved as **WebPublisher Reports** to be used on the Web.

7. Repeat steps 2–6, as necessary.

How Clients Change Forms

Clients can select different forms from a droplist in the Web browser to change the appearance of records. The droplist is generated dynamically when records are displayed after a search, unless you have elected not to permit the user to change forms. (To prohibit the user from changing the forms, see “To omit form droplists” on page 40.)

To make forms appear in the droplist, the textbase designer needs to use the Save Form As dialog box in the Form Designer (choose **Form Operations>Save Form As**). Select the **Report Window** and **WebPublisher Reports** check boxes to make a form appear in the droplist for reports (as shown below). Select the **Record Display Window** and **WebPublisher Reports** check boxes to make a form appear in the droplist for single-record display.

Form droplist —

Name	Item #	Price	Available As	Colors	Status
<u>Nissan 300 ZX</u>	KP3528	\$199.00	Kit Painted	Red	Out of Stock
<u>Porsche</u>	KP1045	\$207.00	Kit Painted	White	In Stock
<u>Porsche</u>	KU1045	\$190.00	Kit Unpainted	White	In Stock
<u>Road Boss</u>	KP2023	\$220.00	Kit Painted	Yellow	In Stock
<u>Road Boss</u>	KU2023	\$194.00	Kit Unpainted	Yellow	In Stock

To specify the location of the droplist

1. In the appropriate designer, open the query screen or menu screen that will pass a query to the textbase.
2. Choose **Tools>Screen Properties** to open the Screen Properties dialog box.
3. On the HTML tab, select **Top of page**, **Bottom of page**, or **Top and bottom of page** from the **Next/previous buttons** droplist to specify where the form droplist and the next record and previous record buttons will appear. This setting affects all *WebPublisher* forms used when a query is submitted through this screen. It also controls where the optional **New Search** button will appear on the report page (even if the next/previous record buttons and form droplist do not appear).

To omit form droplists

1. In the appropriate designer, open the query screen or menu screen that will pass a query to the textbase.
2. Choose **Tools>Screen Properties** to open the Screen Properties dialog box.
3. On the HTML tab, clear the **Show form list on report page** check box to omit the report form droplist and/or the **Show form list on expanded display page** check box to omit the expanded display form droplist. These properties affect all *WebPublisher* forms used when a query is submitted through this screen.

Previewing HTML Reports

To test report forms for Web use, choose **File>Write Report to File**. Write a report to HTML using each form; then view the page in your Web browser. You can test the appearance of report forms this way, without involving the server. This is not a substitute for testing with *WebPublisher*. Only *WebPublisher* shows navigation buttons, form droplists, “See Also” links, and expand record links.

Important! The Write Report to File function does not use Cascading Style Sheets. For more information about Cascading Style Sheets, see “Cascading Style Sheets and Absolute Positioning in Forms” on page 37.

To write a report to HTML

1. Use DB/TextWorks to retrieve one or more records. When you write a report to HTML, the entire report becomes one HTML page. To generate multiple, smaller pages, do searches that generate fewer records, and create each report separately.
2. Choose **Display>Select Forms** to open the Select Forms dialog box.
3. Click the **Report Window** button and specify the form you want to test in the Select Report Form dialog box.
4. Click **OK** to close the Select Report Form dialog box, then click **OK** to close the Select Forms dialog box.
5. Choose **File>Write Report to File** to open the Write Report to File dialog box.
6. Select the **HTML** option button and click **OK**.
7. When prompted, specify a file name and location for the HTML page in the Save File As dialog box, and click **Save**.

Forms for Printed Reports

Printed reports use whichever form is selected in the Web browser when the browser Print command is used.

Query Screens

You can use DB/TextWorks to create query screens and export them to HTML to use on the Web. A query screen consists of one or more boxes, each of which searches one or more fields in a textbase. Web users can point a browser to the HTML page and submit a query. Records that match the criteria will be retrieved and displayed.

Query form for locating company documents

Description of document
AND marketing lit*

Document contents
AND web / internet

File
NOT brochure


Submit Query Reset Help

Retrieval software: DB/Text WebPublisher, provided by INMAGIC

A single query screen can also be used to search multiple textbases. To see how to do this, see “Searching Multiple Textbases from a Query Screen” on page 49.

To design a query screen for Web use

Note: The *Inmagic DB/TextWorks User’s Manual* and DB/TextWorks online help are your best sources for information about designing query screens, but the instructions below can help you get started quickly.

1. Start DB/TextWorks and open the textbase that you will be publishing.
2. Choose **Search>Design Query Screen** to open the Open Query Screen dialog box.
3. Select an option from the Start With list and click **OK**. Now you are working in the Query Screen Designer. For help, click the Help on this Window button  on the Query Screen Designer toolbar.

4. Add boxes to your query screen. A query screen designed for Web use can consist of three types of boxes: query boxes, text boxes, and picture boxes.

Note: While you can add script input boxes and script buttons to query screens, they are ignored when the query screen is exported to HTML because they are for desktop use only. This is because screen scripts are ignored in *WebPublisher*. Sets boxes are also ignored in *WebPublisher*.

- **Query or text boxes.** Choose **Edit>Add>Query Box** or **Edit>Add>Text Box**. Use the Box Properties dialog box to add fields and/or text to the box, as appropriate.
- **Picture box.** Choose **Edit>Add>Picture Box**. Use the Picture Box Properties dialog box to designate the image file you want to include in the query screen. Be sure you place the image in a location where *WebPublisher* will look for it. For more information about images, see “Choosing the Best Image Display Method” on page 59 and “Logo and Background Image File Locations” on page 71.

Note: To edit query and text box content and appearance, use the Box Properties dialog box (choose **Tools>Box Properties**). For example, select the Position tab in either of the Box Properties dialog boxes to move or resize boxes.

5. [Optional] To make each query box only one line high, choose **Edit>Select All>Query Boxes**. Then choose **Tools>Box Properties>Position** and change the **Maximum height** setting to 1. If you do not do this, each query box will be five lines high in the browser and will have scroll bars.
6. Specify Web-specific properties, as applicable, on the Browse Choices and HTML tabs (choose **Tools>Box Properties**). Options on the Browse Choices tab apply to query boxes, those on the HTML tab apply to text boxes. For more information about these properties, see “HTML Properties for Query Screens” on page 44.
7. Choose **Tools>Screen Properties** to open the Screen Properties dialog box and specify properties that apply to the query screen as a whole. For example, you may want to choose a background color for the query screen (General tab).

Important! Pay special attention to the Logos and HTML tabs on the Screen Properties dialog box. You use these tabs to select important Web-specific properties, as explained in the next section, “HTML Properties for Query Screens.”

8. To export the query screen to HTML, choose **Screen Operations>Export Query Screen to HTML**. You can put the HTML page in any location accessible to the HTTP server. If you need help doing this, see “Exporting Query Screens to HTML” on page 52.
9. Choose **Screen Operations>Save Query Screen As**. Name the screen when prompted (clients will not see this name) and save the screen in the textbase file.

Note: Saving a query screen is optional (but recommended) and can be done before or after exporting to HTML.

HTML Properties for Query Screens

HTML properties can apply not only to query screens but also to individual boxes within query screens. When you design query screens for the Web, you can select Web-specific properties from the menu locations listed below.

- **Tools>Screen Properties>Logos**
- **Tools>Screen Properties>HTML**
- **Tools>Box Properties>Browse Choices** (query boxes)
- **Tools>Box Properties>HTML** (text boxes)
- **Tools>WebPublisher Multiple Textbase Query**

These properties only affect query screens exported to HTML for use on the Web. They are ignored for query screens used on the desktop. Each HTML feature is summarized below. For more information, click the **Help** button on each dialog box, as appropriate.

Adding Logos to a Query Screen

Use the Logos tab on the Screen Properties dialog box (choose **Tools>Screen Properties>Logos**) to add images to the top and/or bottom of a query screen.

Type the image name in the **Leading Logo** and/or **Trailing Logo** box, or click the **Browse** button to navigate to it. For more information, see “Choosing the Best Image Display Method” on page 59 and “Logo and Background Image File Locations” on page 71.

Adding HTML Properties to a Query Screen

Use the HTML tab on the Query Screen Properties dialog box (choose **Tools>Screen Properties>HTML**) to add the following HTML-related properties to a query screen:

- **Show Boolean operators.** Select this check box to include a Boolean droplist (**AND**, **OR**, **NOT**) for all searchable boxes on the HTML query screen.
- **Add an ‘Enter password:’ box.** Select this check box to include a box in which the client can type a password. For more information, see “Passwords” on page 25.
- **Use background image.** Select this check box to specify an image file for the page background. For more information, see “Choosing the Best Image Display Method” on page 59 and “Logo and Background Image File Locations” on page 71.
- **Initial Report and Display Forms.** Click these buttons to specify the report form and/or display form to be used by the Web browser to display records. For more information, see “Specifying Initial Forms” on page 38.
- **Show form list on report page/Show form list on expanded display page.** Select these check boxes to specify whether the list of available forms is displayed on report pages or expanded display pages.

- **Records per page.** Type the number of records you want displayed on one report page after a search, or select **unlimited** from the **Records per page** droplist. Limiting the number of records displayed on a page lets users view the records matching their search criteria in batches.
- **Next/previous buttons.** Use the **Next/previous buttons** droplist to specify where the next record and previous record buttons will appear on the report page. The placement option also controls where the form droplists and optional **New Search** button will appear on the report page.

Adding Browse Choices Options to Query Boxes

The Browse Choices tab contains the following properties that relate specifically to HTML:

- **Do not allow browsing.** Select this option button to designate that users cannot browse field entries, but can still search the field by typing a word, phrase, or term.
- **Add Word Wheel.** Select this option button to add a Word Wheel button to the right of the query box. It will appear in the Web browser. Clicking the button opens a dialog box that lists all indexed terms or words in the current field. The user can paste terms or words from the list into the query box.
- **Add choices droplist.** Select this option button to add a droplist (also called a drop-down list) to the selected box. The droplist will appear in the Web browser. When a user clicks the down arrow on the droplist, a list of indexed terms appears. (If there is no Term index, you see a list of words.) The user can select an item from the list and search for that item.

For more information about properties on the Browse Choices tab, see “Eliminating Trial-and-Error Searching” on page 47.

Adding Links to Query Screens

Use the HTML tab on the Box Properties dialog box (choose **Tools>Box Properties>HTML**) to specify how the text you enter in a text box will be interpreted.

Select one of the following options from the **Treat text as** droplist.

- **Do not alter.** Use this option for text boxes that contain information that you do not want to be interpreted as HTML by the client’s Web browser, such as titles and instructions. The text will appear in the browser exactly as it appears in the text box. The Web browser will not attempt to interpret any characters as HTML tag delimiters.
- **Raw HTML.** Use this option to pass the text box content through to the Web browser. When the query screen is used on the Web, the browser will interpret the text as HTML. Type the tagged HTML in the **Text** box on the Text tab of the Text Box Properties dialog box. For example:

```
<a href="http://www.myco.com/adv.htm">Click for advanced screen</a>
```

appears on the query screen as:

[Click for advanced screen](http://www.myco.com/adv.htm)

- **URL.** Use this option if the text is a single URL (for example, `http://www.inmagic.com`). DB/Text *WebPublisher* will add the necessary HTML tags to turn this into a hypertext link.
- **E-Mail link.** Use this option for a text box that contains a single EMail address. *WebPublisher* will interpret it as a “mailto” link. For example:
`joe@mydomain.com`
 will be treated as:
`joe@mydomain.com`
- **HTML file reference.** Use this option to treat the content of a text box as a file reference (file://). This option is commonly used to launch an application on an intranet. For example:
`\\server\volume\info.doc`
 is treated by *WebPublisher* as:
`\\server\volume\info.doc`
 To display alternate text on the screen, in place of the file reference, select the **Use alternate link text** check box. Type the alternate text (for example, `Click here`) in the **Text** box. Users will see `Click here` on the screen. For more information about using HTML file references to launch an application, see “Launching Applications on an Intranet” on page 61.

Adding Multiple Textbase Searching Options to a Query Screen

If you want to search more than one textbase with a single query screen, use the options on the WebPublisher Multiple Textbase Query dialog box.

- **Lead Textbase for WebPublisher Multiple Textbase Query.** Select this check box to specify the textbase you have open as the lead textbase for searching multiple textbases on the Web. For more information about multiple-textbase searching, see “Searching Multiple Textbases from a Query Screen” on page 49.
- **Textbases to Search.** Click the **Add** button to specify additional textbases you want to search simultaneously with the lead textbase. The names of the additional textbases you select appear in the Textbases to Search list.
- **Initial Report Form/Initial Display Form.** Click these buttons to specify the report form and/or display form to be used by the Web browser to display records for the textbase selected in the Textbases to Search list.
- **Create Field Name Map.** Click this button to map fields in the lead textbase to fields in the textbase selected in the Textbases to Search list for the purpose of multiple textbase searching.

Eliminating Trial-and-Error Searching

To make searching easier and more accurate, you can give Web clients the ability to select information from a list to use as search criteria. Do this to selected boxes on a query screen by adding a choices droplist or a Word Wheel.

Choices Droplist

A choices droplist (also known as a drop-down list) is a static list of indexed items in a field. The list is generated when you export the query screen to HTML. Droplists can be seen in Web browsers but not on the desktop. Clients can select an item from the list instead of typing query criteria. If clients do not want to search that field, they can select a blank line from the top of the droplist.

Droplists are appropriate for fields that contain short, unchanging lists (for example, two to 15 items), as shown in the illustration below. In addition, droplists work best with lists whose items are limited to fewer than 50 or so characters each. This is because a droplist automatically sets its width to the width of the longest entry in the list, to a maximum of 250 characters. Avoid using droplists for lengthy fields—use a Word Wheel instead.

The screenshot shows a web form titled "Type Search Criteria of Select from a List". It contains three search criteria sections: "Size", "Color", and "Notes". Each section has a dropdown menu with "AND" selected. The "Color" dropdown is open, showing a list of colors: blue, green, red, and yellow. Below the dropdowns are "Submit Query" and "Reset" buttons, and a "Help" icon. A text box to the right of the form says: "Users can open the droplist to see (and search for) actual indexed information."

A droplist shows entries from the Term index. If there is no Term index, the droplist shows entries from the Word index. You cannot add a droplist to an unindexed field. If a box searches multiple fields, the droplist shows entries only from the first field in the box. (To re-order fields, select a query box, choose **Tools>Box Properties>Fields** and use the **Change Order** buttons.)

To add a choices droplist

1. Choose **Search>Design Query Screen** and open a query screen in the Query Screen Designer.
2. Select the query box to which you want to add a choices droplist.

3. Choose **Tools>Box Properties** to open the Query Box Properties dialog box.
4. On the Browse Choices tab, select the **Add choices droplist** option button. To limit the number of entries that will appear, select the **Specify number of entries** check box, then type the maximum number of entries you will allow. For example, if you type 15, only the first 15 entries will appear in the list.

Note: The default and maximum number of terms that can appear is 200.

5. Click **Apply**, the **Close**.
6. Save the query screen, then export it to HTML (see page 52 if you need help doing this). When you export the query screen to HTML, a droplist will be generated as a static list and saved as part of the HTML page. If the textbase changes after the query screen is exported, you should re-export the query screen to update the list, or manually edit the HTML file.

Word Wheel

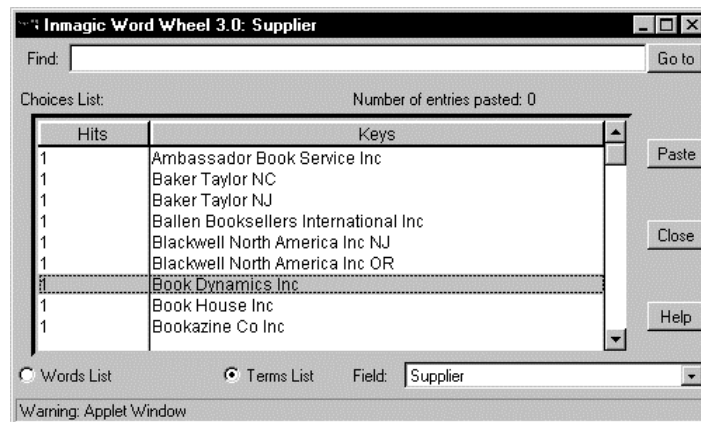
A Word Wheel is a dialog box that shows indexed information from the textbase. When you design a query screen, you can add a Word Wheel button next to a box, so it appears like this in the browser:

Supplier

	Word Wheel
--	------------

Clients can click the **Word Wheel** button to open a dialog box that shows information from the textbase. *WebPublisher* generates Word Wheels dynamically, so they always show the most recent information. The Inmagic index-streaming technology downloads only the portion of the index that is needed, to minimize load on the server.

The Word Wheel can show a list of words or terms (complete entries). The client selects a word or term from the Keys list, clicks **Paste**, then executes the query.



Select a term from the list, then click the **Paste** button.

If the query box contains multiple fields, select the field for which you want to see the Choices List from the **Field** droplist.

To add a Word Wheel button

1. Choose **Search>Design Query Screen** and open a query screen in the Query Screen Designer.
2. Select the query box to which you want to add a Word Wheel.
3. Choose **Tools>Box Properties** to open the Box Properties dialog box.
4. On the Browse Choices tab, select the **Add Word Wheel** option button. By default, the Word Wheel will show a list of words. To show terms first, select the **Show terms by default** check box.

Note: Users can switch between seeing words and terms by selecting the **Words List** and **Terms List** option buttons, as appropriate, on the Word Wheel dialog box.

5. Click **Apply**, then **Close**.
6. Save the query screen, then export it to HTML (see page 52 if you need help).

Important! See Chapter 4, “Client Requirements and Resources” on page 65 for browser requirements to run the Word Wheel. If you suspect that some clients will have older browsers, you may want to provide access to two versions of the search screen—one with Word Wheels and one without.

Searching Multiple Textbases from a Query Screen

When you publish textbases on the Web with DB/Text *WebPublisher*, you can design a query screen so users can search more than one textbase with it.

Searching multiple textbases with one query screen is helpful when users are looking for related information that is kept in different textbases. For example, an attorney at a large law firm can simultaneously search the textbases at all of the branch offices for information on a particular client or topic. Corporate librarians can use the multiple-textbase search feature to search their book and journal textbases simultaneously for a particular author or subject.

To search multiple textbases using a single query screen, you must specify one textbase as the “lead textbase,” meaning it contains the query screen definition with all of the pertinent information about the other textbases to be searched. Then you must map fields between the lead textbase and the other textbases. Typically, the textbases being searched are similar in structure, with similar fields. You can also specify which report and display forms are used for each textbase.

To set up a *WebPublisher* query screen to search multiple textbases

1. Open the textbase that you want to specify as the lead textbase.

Note: The textbase chosen as the lead textbase is the one with the records you want to appear first in reports (for example, the local book catalog before branch library book catalogs). It may also have secondary fields you want to search, or be the textbase with the index you want displayed in a Word Wheel. A Word Wheel can only be used to show the index for the lead textbase.

2. In the Query Screen Designer, open the query screen from which you want to search multiple textbases. You can use an existing *WebPublisher* query screen, or design a new one.
3. Choose **Tools>WebPublisher Multiple Textbase Query** to open the WebPublisher Multiple Textbase Query dialog box.
4. Select the **Lead Textbase for WebPublisher Multiple Textbase Query** check box. This specifies the textbase you have open as the lead textbase.
5. Click **Add** to open the Open Inmagic DB/TextWorks Textbase dialog box and specify the textbase you want to search simultaneously from the query screen. Click **Open**. The textbase will appear in the Textbases to Search list in the WebPublisher Multiple Textbase Query dialog box. Do this for each additional textbase you want to search simultaneously from the query screen.

Important! If you plan to have users search multiple textbases from a query screen, you must make sure the paths of all the textbases to be searched are in the DBTWPub.INI file.

6. Select a textbase in the Textbases to Search list and click the **Create Field Name Map** button to open the Create Field Name Map dialog box.
7. Specify which fields will be searched in each textbase when a user submits a query by selecting a field from the Lead Textbase Fields in this Screen list and then a corresponding field from the Fields Available in this Textbase list.

Note: The fields in the Fields Available in this Textbase list are always from the textbase selected in the Textbases to Search list in the WebPublisher Multiple Textbase Query dialog box.

8. Click the **Create Map Between these Selected Fields** button and the two fields appear in the Mapped Fields list. This means that when a user searches the field shown from the lead textbase, the field shown from the other textbase will also be searched.
9. Repeat steps 7 and 8 for each field you want to map between the lead textbase and the textbase selected in the Textbases to Search list. Click **OK** when done.

Note: Unmapped fields will automatically be mapped to the corresponding field in the lead textbase. The corresponding field is the field that is in the same place structurally as the unmapped field (for example, if the first field in the textbase is unmapped, it will be mapped to the first field in the lead textbase).

10. [Optional] To delete fields that have been mapped, select them in the Mapped Fields list, then click the **Delete Entry** button. Use the **Clear All Entries** button to delete all mapped fields in the list.

Tip! Notice that a message telling you how many fields are mapped appears next to the **Create Field Name Map** button on the WebPublisher Multiple Textbase Query dialog box.

11. Use the **Initial Report Form** and **Initial Display Form** buttons to specify the default forms used when results are displayed from the textbase selected in the Textbases to Search list. Notice that the names of the form appear next to the applicable button.

Note: The default forms for the lead textbase are specified in the query screen (choose **Tools>Screen Properties>HTML**).

12. Repeat steps 6–11 for each textbase in the Textbases to Search list.
13. Click **OK**.
14. Save the query screen and export it to HTML, as explained on page 52.

Important! When saving a query screen that will search multiple textbases, you must select the **Textbase File (Public)** option button for it to be used on the Web.

How multiple textbase query results are displayed

When a user searches with a query screen designed to search multiple textbases, they use it the same way as one designed to search a single textbase.

Records retrieved from the lead textbase appear first, in the default report and display forms specified in the lead textbase.

Records from subsequent textbases searched appear next, in the order in which those textbases were added in the Textbases to Search list. Results from each of those textbases appear in the report and display forms specified on the WebPublisher Multiple Textbase Query dialog box for that textbase.

Tip! If you want the results of a multiple-textbase search to appear uniform, design identical forms for each of the textbases being searched and specify them as the report and display forms.

Consider using forms that show all fields applicable to your users, as users can not change forms when using a multiple-textbase query.

Exporting Query Screens to HTML

To use a query screen on the Web, you must export it to HTML.

To export a query screen to HTML

1. Open the query screen in the Query Screen Designer.
2. Choose **Screen Operations>Export Query Screen to HTML**. Name the file when prompted, and indicate where to save it. You can put the HTML page in any location accessible to the HTTP server (it does not have to go in any particular directory).
3. Click **Save**.

During the export, the box labels, the list of fields to be searched by each box, and the Boolean command associated with each box are preserved. Buttons labeled **Submit Query** and **Reset**, and a **Help** icon are added automatically. Sets boxes, script input boxes, and script buttons are not applicable to the Web and will not appear. Horizontal box position, label position, font faces, and Tab order do not translate to the Web.

If you make changes to a query screen in the Query Screen Designer after it has been exported to HTML, you must re-export it for your changes to be seen on the Web.

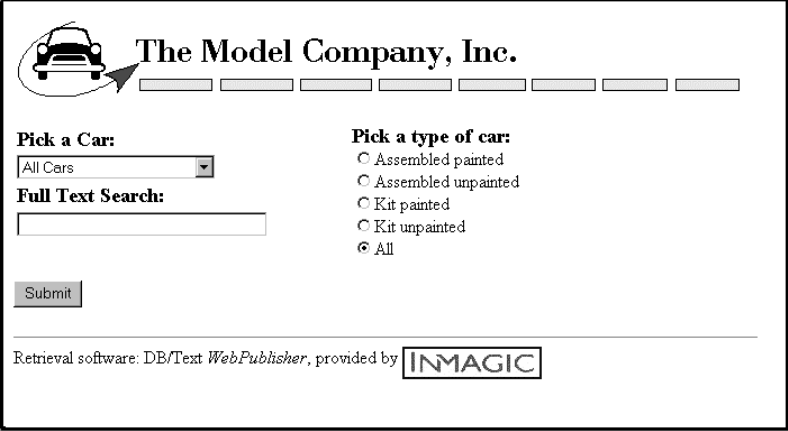
Tip! You can use HTML code to format boxes on your query screen. Type the HTML code you want in a text box and be sure **Raw HTML** is selected from the **Treat text as** droplist on the HTML tab (choose **Tools>Box Properties>HTML**). For example, you can use HTML to center boxes.

Editing HTML Query Screens


You can use standard HTML editing tools to enhance your search screens. After exporting a query screen to HTML in DB/TextWorks, open the file in the HTML editor of your choice and add the features you want, such as check boxes and option buttons. You probably should keep related buttons, inputs, and applets near each other on the page so as not to confuse the user. As with any HTML form, be sure to test it thoroughly before putting it up on the Web.

Important! If you re-export an edited query screen to HTML using DB/TextWorks, any changes made outside of DB/TextWorks will be lost.

The following illustration shows a search screen called CARSXTRA.HTM, which is included with DB/Text *WebPublisher* in the SAMPLE subdirectory. You can open this page in your Web browser and view the document source. With a little practice, you can achieve similar effects for your own search screens.




The screenshot shows a web form for 'The Model Company, Inc.' with a car icon and a search interface. It includes a 'Pick a Car' dropdown menu, a 'Full Text Search' input field, a 'Submit' button, and a 'Pick a type of car' section with radio button options. At the bottom, it mentions 'Retrieval software: DB/Text WebPublisher, provided by IMAGIC'.

 **The Model Company, Inc.**

Pick a Car:

Full Text Search:

Pick a type of car:
☐ Assembled painted
☐ Assembled unpainted
☐ Kit painted
☐ Kit unpainted
☒ All

Retrieval software: DB/Text *WebPublisher*, provided by 

Menu Screens

A menu screen is a list of pre-defined queries that appear as hypertext links. You can use DB/TextWorks to create menu screens and export them to HTML to use on the Web. Menu screens are especially useful when you do not want clients to have to specify query criteria, and when many clients are likely to want to perform the same searches. For example, the following page includes three pre-defined queries for commonly requested information.



Menu screens used on the Web are quite different from menu screens used on the desktop with DB/TextWorks. Each menu screen for the Web contains at least one “initial action,” which is a pre-defined search.

Tip! It is especially useful to use saved queries including @DATE as the initial action. For example, you could have an initial action (query) called *Upcoming Events*. The query for this action could be @DATE : @DATE+14. This query would return all event records from the date of the query through the next two weeks.


When the page is displayed in a browser, the searches look like regular links. When a client clicks on a link, *WebPublisher* performs a query and displays records in the browser. From the user’s point of view, a regular hypertext jump has occurred. It is not immediately evident to the user that a textbase has been queried.

Important! When the link is clicked, *WebPublisher* re-executes the query. *WebPublisher* does not simply load the set. The set will include any omitted records and any new or changed records that now match the query criteria.

To design a menu screen for Web use

The *Inmagic DB/TextWorks User's Manual* and online help are your best sources for information about designing menu screens, but the instructions below can get you started quickly.

Note: Before you begin, create one or more pre-defined queries that you will associate with items on the menu screen. Start DB/TextWorks, open a textbase, and perform a query. After retrieving records, choose **Sets>Save Set As**. When prompted, name the set and save it in the **Textbase File (Public)**. The search criteria will automatically be saved as a saved query. Web clients will not see the name you assign. Repeat this step for each search that you include on the menu screen.

1. Start DB/TextWorks. Do not open a textbase. If a textbase is open, close it (choose **File>Close**).
2. Choose **Menu Screens>Design** to open the Open Menu Screen dialog box.
3. Select an option button to specify whether to open the current menu screen, a new one, or an existing one. If you are creating a new menu screen, name it when prompted and specify its location. For more information, click the Help on this Window button  on the Menu Screen Designer window.
4. Choose **Edit>Add>Textbase Box** to add a textbase box. The Textbase Box Properties dialog box opens.
5. Specify the attributes you want for the menu screen by using the tabs on the Textbase Box Properties dialog box:
 - **Contents tab.** Click the **Specify Textbase** button and select the textbase you want to be searched. In the **Description** box, type the text you want to appear on the menu screen as hypertext. This is what clients will click on to initiate the search. For example, type *Upcoming Events*.
 - **Position tab.** Specify box size and position.
 - **Font, Color tab.** Specify the font size.
 - **Icon tab.** The options on this tab are for desktop use only and are not applicable to menu screens used on the Web.
 - **Initial Elements tab.** Click the **Specify Initial Elements** button and specify the default forms you want to use for menu screen. You only need to specify a Report Window form (for reports) and a Record Display form (for expanded display). The other initial elements only apply to the desktop.
 - **Initial Action tab.** Select the **Saved query** option button to enable the **Specify Saved Query** button. Click the button and specify which pre-defined query to use on the menu screen. (Choose one of the saved queries you created at the beginning of this topic.)
6. Repeat steps 4 and 5 for each search that you want to include on the menu screen.

7. [Optional] Add a title or instructions by choosing **Edit>Add>Text Box** to open the Text Box Properties dialog box. Type the text you want in the **Text** box, or click the **Enlarge** button to open a larger, resizable text editor dialog box instead. When you click **OK** on the text editor dialog box, your text appears in the **Text** box on the Text tab automatically. For example, type `Welcome to Beantown!` in the **Text** box.

Note: You can add Web-specific properties to text boxes on a menu screen using the options on the HTML tab of the Text Box Properties dialog box. For more information about these properties, see “HTML Properties for Menu Screens” on page 57.

8. [Optional] Add a picture box by choosing **Edit>Add>Picture Box**. Use the Picture Box Properties dialog box to designate the image file you want to include in the menu screen. Be sure you place the image in a location where *WebPublisher* will look for it. For more information about displaying images, see “Choosing the Best Image Display Method” on page 59 and “Logo and Background Image File Locations” on page 71.
9. If needed, edit the content and appearance of textbox boxes, text boxes, and picture boxes. Here are some guidelines to help you get started:
 - You must select a box to be able to make changes to it.
 - Use the Edit menu to add, copy, and delete boxes.
 - Use **Tools>Box Properties** to change the attributes of the selected box.
 - You can use an HTML editor to make changes after exporting a menu screen to HTML.
10. [Optional] Specify properties that apply to the menu screen as a whole by choosing **Tools>Screen Properties** to open the Screen Properties dialog box. For example, you may want to choose a background color for the menu screen by specifying it on the General tab. Note that the **Path settings** option on the General tab does not apply to *WebPublisher* menu screens.

Note: Use the Logos and HTML tabs on the Screen Properties dialog box to select Web-specific properties, as explained in the next section.

11. Export the menu screen to HTML by choosing **Screen Operations>Export Menu Screen to HTML**. If you need help doing this, see “Exporting Menu Screens to HTML” on page 58. You can put the HTML page in any location accessible to the HTTP server.
12. Save the menu screen by choosing **Menu Operations>Save Menu Screen As**. Specify a name and location when prompted. The location does not have to be accessible to the server because Web clients will not see this file. Do not type a file extension, because .TBM is added automatically.

Note: Saving a menu screen is optional (but recommended) and can be done before or after exporting to HTML.

HTML Properties for Menu Screens

HTML properties can apply to both menu screens and individual boxes on menu screens. When you design menu screens for the Web, you can select Web-specific properties from the menu locations listed below.

- **Tools>Screen Properties>Logos**
- **Tools>Screen Properties>HTML**
- **Tools>Box Properties>HTML** (for text boxes only)

These properties only affect menu screens exported to HTML for use on the Web. They are ignored for menu screens used on the desktop. For more information, click the **Help** button on the Screen Properties and/or Text Box Properties dialog boxes.

Adding Logos to a Menu Screen

Use the Logos tab on the Screen Properties dialog box (choose **Tools>Screen Properties>Logos**) to add images to the top and/or bottom of a menu screen.

Type the image name in the **Leading Logo** and/or **Trailing Logo** box, or click the **Browse** button to navigate to it. For more information, see “Choosing the Best Image Display Method” on page 59 and “Logo and Background Image File Locations” on page 71.

Adding HTML Properties to a Menu Screen

Use the HTML tab on the Menu Screen Properties dialog box (choose **Tools>Screen Properties>HTML**) to add the following HTML-related properties to a menu screen:

- **Use background image.** Select this check box to specify an image file for the page background. See “Choosing the Best Image Display Method” on page 59 and “Logo and Background Image File Locations” on page 71.

Note: To specify a color for the menu screen background, choose **Tools>Screen Properties>General**.

- **Show form list on report page/Show form list on expanded display page.** Select these check boxes to specify whether the list of available forms is displayed on report pages or expanded display pages.
- **Records per page.** Type the number of records you want displayed on one report page after a search, or select **unlimited** from the **Records per page** droplist. Limiting the number of records displayed on a page lets users view the records matching their search criteria in batches.
- **Next/previous buttons.** Use the **Next/previous buttons** droplist to specify where the next record and previous record buttons should appear on the report page. It also controls where the form droplists and optional **New Search** button appear on the page.

Using HTML Properties in Text Boxes on a Menu Screen

Use the HTML tab on the Box Properties dialog box (choose **Tools>Box Properties>HTML**) to specify how the text you enter in a text box on a menu screen will be interpreted. Select an option from the **Treat text as** droplist. These options are the same as for query screens. For more information, see “Adding Links to Query Screen” on page 45.

Exporting Menu Screens to HTML

To use a menu screen on the Web, you must export it to HTML.

Before exporting, you should first specify default forms for each textbase box (choose **Tools>Box Properties>Initial Elements**) and whether form droplists should appear on report and/or display pages (choose **Tools>Screen Properties>HTML**).

To export a menu screen to HTML

1. Open the menu screen in the Menu Screen Designer.
2. Choose **Menu Operations>Export Menu Screen to HTML**.
3. In the Save File As dialog box, name the file and specify its location when prompted. The location should be accessible to the Web server.
4. Click **Save**.

The only items that are exported are textbase boxes whose Initial Action is a saved query, text boxes, and picture boxes. Textbase boxes that do not have a saved query specified as the Initial Action will not translate to the Web.

In addition, script input boxes and script buttons are not applicable to the Web and will not appear. Horizontal box position, label position, font faces, and Tab order do not translate to the Web.

Editing HTML Menu Screens

You can use standard HTML editing tools to enhance your menu screens. After exporting a menu screen to HTML in DB/TextWorks, open the file in the HTML editor of your choice and add the features you want. As with any HTML form, be sure to test it thoroughly before putting it up on the Web.

Note: If you re-export an edited menu screen to HTML using DB/TextWorks, any changes made outside of DB/TextWorks will be lost.

Images

You have a great deal of flexibility about where to place image files, but you will need to coordinate image file locations with the Webmaster, as explained in Chapter 5, “For the Webmaster,” on page 67.

Choosing the Best Image Display Method

When placing an image in a form, query screen, or menu screen designed for Web use, you can choose different methods to display the images. The methods you choose depend upon whether the images are fixed images or record images. Fixed images are used in forms, query screens, and menu screens; record images are used in forms only.

All images, except for those being used in picture boxes, should be sized appropriately. For example, you may want to reduce the dimensions of some images to fit in a tabular form. Very large images may work better as links.

Displaying Fixed Images

You can use fixed images in forms, query screens, and menu screens. They can appear not only in the body of the form or screen, but also at the top, bottom, and/or as the background of the form and/or screen. When displaying fixed images, keep in mind the purpose of the image and where you want to place it, as these considerations determine the type of fixed image you use.

There are three types of fixed images:

- **Picture box image.** Use a picture box to add an image in the body of a report or screen. Use a picture box if you want the ability to change the image size, or if you want to place the image in the body of the page.
- **Logo.** Use a logo when you want to place an image file at the top and/or bottom of a form, query screen, or menu screen. Logos are placed above and below the controls added automatically by *WebPublisher*, such as the **Next Record** and **Previous Record** buttons. Images placed at the top of a form or screen are leading logos; images placed at the bottom are trailing logos. The actual size of the image is used.
- **Background image.** Use a background image when you want to specify an image file as the page background for a form, query screen, or menu screen.

Displaying Record Images

You can use record images in forms. When displaying record images, keep in mind that some fields contain multiple images, so it is important to choose the method that will display the number of images you want, in the correct format. Record images appear in the body of the record.

There are three types of record images:

- **Picture box image.** Use a picture box when you only want to show the first image in a field, in a box by itself, or when you want the ability to change the size of the image. Using a picture box for record images retains Cascading Style Sheets and requires that the user have HTTP access. For more information, see “Cascading Style Sheets and Absolute Positioning in Forms” on page 37.
- **Inline image.** Use an inline image when you want to display all images in a field that contains multiple images, display images in the same box with other content (such as text), and/or display an image not contained in an Image field. Inline images can be called by a file reference or an HTTP reference. The actual size of the image is used.

Note: When inline images are used, *WebPublisher* uses simple HTML.

- **Image link.** Use an image link when you want to display a link to the image, rather than the image itself, because the image is unusually large (for example, file-size and/or dimensions). Inline images can be called by a file reference or an HTTP reference. The actual size of the image is used.

For more information about using inline images and image links, see “Using File References for Images on an Intranet” on page 63. For more information about picture boxes, see the online help.

Important! When entering file names in an Image field for Web use, you must enter each file name individually. For example, IMAGE001.JPG, IMAGE002.JPG, IMAGE003.JPG. When entering file names in an Image field for desktop use only, you can use a shortcut naming convention (for example, IMAGE001.JPG 3). For more information about Image file names and naming conventions, see the DB/TextWorks online help.

Textbase Record Images

A textbase may include fields that hold the names of image files. For example, a field called *Schematics* might contain the entry DIAGRAM1.JPG.

Most Web browsers support only JPEG and GIF image formats. JPEG images can be viewed in both Web and desktop environments without any plug-ins or other software. GIF files can be viewed on the Web but not on the desktop with DB/TextWorks. If you are sharing a textbase between desktop users and Web clients, use JPEG images if possible, or ensure that your Web clients have an appropriate image display plug-in.

Intranets

You can use DB/Text *WebPublisher* on an intranet as well as on the Internet. An intranet is a local or wide-area network that operates in a manner similar to the Internet, using TCP/IP, browsers, and HTML pages. By publishing textbases on an intranet, you can make information widely available within your organization across a variety of platforms. For example, you can publish company policies and procedures and provide shared access to image files.



Publishing textbases on an intranet is identical to publishing on the Internet, except that you can launch applications and access images using `file://` references in addition to, or instead of, HTTP.

Important! Be sure to check with the Webmaster before using network file access, to find out which protocol is preferred.

Launching Applications on an Intranet

On a suitably configured intranet, you can launch applications from a form, query screen, or menu screen. For example, a link on a report could open a document in Word or run a video clip. Web clients need network access to the necessary application(s) and files. Be sure to communicate this to the Webmaster in case there are firewall or security issues. For more information, see Chapter 5, "For the Webmaster," on page 67.

The target file on the server should be accessible to all clients, either by HTTP addressing protocols or by network file access protocols, such as Sun NFS or Novell NetWare. Check with the Webmaster to find out which protocol to use.

If you will use HTTP access, specify a link as you normally would, using the **Treat content item as** or **Treat text as** droplist (choose **Tools>Box Properties>HTML** for a form or text box) in the appropriate designer.

If you will use network file access, use the following steps to make *WebPublisher* treat the content of a box as a file reference.

To treat box content as a file reference

1. Add the destination file name in the appropriate designer:
 - Form Designer
 - ◆ Choose **Edit>Add>Form Box** to add a form box and open the Form Box Properties dialog box.
 - ◆ On the Fields subtab of the Contents tab, add a field that contains a file name, such as a field called *Documents*.
 - Query Screen Designer or Menu Screen Designer
 - ◆ Choose **Edit>Add>Text Box** to add a text box and open the Text Box Properties dialog box.
 - ◆ On the Text tab, type the full path name for a document. For example, type `\\server\vol1\test\myfile.doc`.
2. With the Form/Text Box Properties dialog box still open, select the HTML tab.
3. Select **HTML file reference** from the **Treat content item as** (form box) or **Treat text as** (text box) droplist. Click **Apply**.

To avoid conflicts over logical drive mappings, specify the target file using a UNC file name, not a drive letter. If you want the link text to be different than the file name, select the **Use alternate link text** check box on the HTML tab, which is enabled when you choose **HTML file reference** from the **Treat content item/text as** droplist. Type the text you want to use as the link in the Text box. For example:

File name:

`\\server\vol1\test\myfile.doc`

Alternate text:

Click here

Result in report:

`Click here`

Using File References for Images on an Intranet

You can access textbase record image files (images referenced in a textbase) on an intranet the same way as you would on the Internet. However, you also have the option of accessing the images by means of a network file addressing protocol, such as Sun NFS or Novell NetWare. The advantage of this approach is that both DB/TextWorks and DB/Text *WebPublisher* can access the images in a shared location, by using the same file access protocol. (Compare this to Internet access, where *WebPublisher* uses HTTP and DB/TextWorks uses file access, so sharing image files is not always practical.)

To use file references for textbase record images

1. In the Form Designer, choose **Edit>Add>Form Box** to add a form box and open the Form Box Properties dialog box.
2. On the Fields subtab of the Contents tab, select a field that contains an image file.

Note: To ensure that everyone has access to the images, the UNC file name must be specified (for example, `\\server\vol1\images\poe.jpg`).

3. On the HTML tab, select **Inline image** or **Image link** from the **Treat content item as** droplist. Selecting **Inline image** displays the image as part of the form, **Image link** creates a link to the image on the form.
4. Select the **Use file:// reference for image** check box, so *WebPublisher* will convert the entry in the field to a file URL.

Note: The “conversion” is done in the form. The textbase records are not modified.

5. Click **Apply**, then **Close**.

If you selected **Inline image** in the steps above, the entry:

`\\server\vol\images\poe.jpg`

will be treated as:

```

```

If you selected **Image link**, it will be treated as:

```
<a href="file://server/vol/images/poe.jpg">\\server\vol\images\poe.jpg</a>
```

Destination

Link Text

Notice that the destination and link text show the file name. You can specify different link text, if you want.

To specify alternate link text

1. In the Form Designer, select a form box and choose **Tools>Box Properties** to open the Form Box Properties dialog box.
2. On the Contents tab, select the field containing the image file name from the Contents list.

3. Select **HTML file reference** from the **Treat content item** as droplist.
4. Select the **Use alternate link text** check box, then specify the text to use as the link in the **Text** box. For example, type `Show Image` to generate this link:

```
<a href="file://server/vol/images/poe.jpg">Show Image</a>
```
5. Click **Apply**, then **Close**.

Note: If there are multiple entries in the field, the alternate link text will be the same for all the entries. For example, if you use `Click here` as the alternate link text and there are three entries in the field, the alternate link text for each entry will be `Click here`.

Testing your Textbases: A Checklist

Before you make your textbases widely available on the World Wide Web or an intranet, test them from a remote computer running a browser (not from the server itself) to be sure that everything works the way you intended. The best way to test is to run through a sample search, retrieval, and display. Use the following list as a checklist of the major areas to test.

- Have you handed off all the necessary files to the Webmaster and coordinated issues about initialization files and textbase and image locations? For more information, see Chapter 5, “For the Webmaster,” on page 67.
- Do all your search screens appear as you intended? Try several queries to see if you retrieve the intended records.
- Do all images appear correctly? Check logos, backgrounds, and textbase record images on forms, query screens and menu screens.
- Do all jumps go to the correct destinations? Verify that there is a jump on your Web site that takes you to the search screen. Check all jumps on query screens, menu screens, and reports.
- Does every report include an expand record link, if appropriate? Is the link text available for every record?
- Do all your reports look the way you intended? Try several different searches, to obtain different record sets. Select each report form and display form in the browser. Different record sets, such as records that lack information in a field, may affect appearance.
- If you made changes to the supplied help pages (WP*.HTM), click the **Help** button on the query screen and in the Word Wheel dialog box to be sure you access the changed help pages. If you renamed the pages, test the inter-page jumps. For example, WPBEGIN.HTM includes links to WP_WW.HTM and WPMSG.HTM.
- Consider doing a usability test to be sure that your search screens and forms are clear and intuitive. Try having an uninitiated user go to your home page and see if that person can figure out what to do: how to jump to a search screen, which fields to query, what to search for, and what syntax to use.

Chapter 4: Client Requirements and Resources

You should be aware of the issues that may have an effect on clients who visit your Web site.

Client Requirements

Except for a Web browser that supports JavaScript/JScript 1.0, Web clients have no system or software requirements. Microsoft Internet Explorer, version 3.0 or later, or Netscape Navigator, version 3.0 or later, is recommended. The textbase designer may want to add a text box to query screens to convey this information and provide links for clients to download more recent versions of browsers.

DB/Text *WebPublisher* automatically detects the browser version.

To display tabular forms or Word Wheels, clients need version 3.0 or later of Internet Explorer or Netscape Navigator. If the browser is version 4.0 or later, Word Wheels will include scroll bars, which make navigation easier, and reports can use Cascading Style Sheets. For more information about CSS, see “Cascading Style Sheets and Absolute Positioning in Forms” on page 37.

How Clients Get Help

DB/Text *WebPublisher* includes several HTML help pages.

- **WPBEGIN.HTM.** Explains how to perform a query and display results. Clients can view this page by clicking the Help icon on an HTML query screen in the Web browser.
- **WP_WW.HTM.** Explains how to use the Word Wheel. Clients can view this page by clicking the **Help** button in the Word Wheel dialog box in the Web browser.
- **WPMMSG.HTM.** Displays an alphabetical list of error messages. Intended primarily for administrators, although you may allow client access. Clients can view this page from links provided on .HTM help pages mentioned above.

The textbase designer is free to make changes to these pages. After making changes, check all links in a browser to make sure everything works as intended.

Query Syntax

Clients can use the following techniques when typing search criteria in a query screen. For more information, read the WPBEGIN.HTM help page supplied with DB/Text *WebPublisher*.

- **Words.** Type a word (for example, `computer`).
- **Phrases.** Type a phrase (for example, `new blue moon`) to find those words, in that order.
- **Word stems.** To find variations based on word stems, type an asterisk at the end of one or more words (for example, `tech*`).
- **Words near each other.** Use `w#` and `p#` between words to do proximity searches. For example, `red w3 boat` finds red within three words of boat (before or after), and `red p3 boat` finds red preceding boat by three words or fewer. You can use an asterisk at the end of either word (for example, `local* w2 network*`).
- **Terms.** Use an equal sign (=) in front of an item to find a complete, exact match, with no additional text before or after. For example, `=america the beautiful` finds that title only. You can do term searches on any Term-indexed field.
- **Dates.** Type a date in any reasonable format, such as: `Dec 99`, `1999 Dec`, `31-Dec-99`, `12-99`, `December 1999`. Do not use a forward slash to separate date elements unless you surround the date with quotation marks ("`12/31/98`") or unless the option to require spaces around search symbols has been set to 1 in the INMAGIC.INI file (`SpacedRelOps=1`). Otherwise, the slashes will be interpreted as Boolean OR symbols rather than as a part of the date. Type `@DATE` to find today's date, or to perform calculations on today's date.
- **Ranges.** Type `1998:2001` to find all dates from 1-1-98 through 12-31-01. If `SpacedRelOps=1` in the INMAGIC.INI file, spaces are required around the colon in order to have it interpreted as a range operator. If `SpacedRelOps=0`, spaces are not required.
- **Comparisons.** Use comparison operators (`<`, `>`, `<=`, `>=`) when searching for dates, values, or text in any Term-indexed field. For example, `<Jan 1998` finds dates before January 1, 1998.
- **Boolean symbols.** Include Boolean symbols (`&`, `/`, `!`) between words, phrases, or terms to represent AND, OR, NOT. For example, `cars & boats` finds records only if they contain both words (records that contain only one of the words are ignored). Type `cars / boats` to find records about either cars OR boats (or both). Type `cars ! boats` to find cars but not boats. Use parentheses to control evaluation order in long queries, such as: `red & (white / blue)`. If `SpacedRelOps=1` in the INMAGIC.INI file, you must include a space before and after the Boolean symbol.
- **Boolean droplists.** If the query screen has Boolean droplists in front of each box, clients can select **AND**, **OR**, **NOT** from the droplist to control how multiple criteria are combined.

Chapter 5: For the Webmaster

Information for Webmasters is consolidated in this chapter. Textbase designers are strongly encouraged to also read this chapter, because they will need to coordinate file locations and other issues with the Webmaster. Before you can use DB/Text *WebPublisher*, you must set up a virtual directory called /DBTW-WPD on the HTTP server, as explained in Chapter 1.

Publishing a DB/TextWorks Textbase

A DB/TextWorks textbase is designed and maintained using DB/TextWorks software running under Microsoft Windows. The information in that textbase can be made available to local or remote Internet or intranet users running browsers, through the use of DB/Text *WebPublisher*.

Here is a list of the tasks involved in publishing a textbase on the Web:

1. Install DB/Text *WebPublisher* on a server running Windows NT or Windows 2000 Server and HTTP (“Web”) server software, such as Microsoft Internet Information Services (IIS).
2. Copy the textbase and its associated files to a directory that can be accessed by DB/Text *WebPublisher* using a file path.
3. Clients point their browsers at a static HTML search page created using DB/TextWorks. This page submits the query to DB/Text *WebPublisher*, which returns the results as a dynamically formatted HTML document to the client’s browser for display.

The two primary tasks involved in publishing a textbase are to store the files in appropriate locations and ensure that appropriate permissions are set for those locations.

During installation, *WebPublisher* creates a directory structure with default locations for the files associated with a textbase. You can copy files to the default directories, or to other directories on the same or a different file server, then use DBTWPub.INI to tell *WebPublisher* where to look.

There are two kinds of “addresses” that *WebPublisher* uses. Sometimes it opens files using a file path. For example:

```
C:\PROGRAM FILES\INMAGIC\WEBPUB\TEXTBASE\MYDIR
```

At other times, *WebPublisher* constructs a URL to pass to the browser, in which case it uses an HTTP-style address. For example:

```
/DBTW-WPD/HELP/WPBEGIN.HTM
```

The relationship between the two kinds of addresses is controlled by the file locations specified in DBTWPub.INI and the mapping settings (also called virtual directories) used by your HTTP server software. To understand the relationship, read “File Locations” on page 69 and “Creating a Virtual Directory” on page 8.

Publishing a “Live” Textbase

The procedures described in this chapter assume that the textbase is being maintained and edited in one location (with DB/TextWorks), then copied to the HTTP server for Web access. Occasionally, you may want to provide Web access to the “live” textbase, to ensure that the most current information possible is being retrieved (so DB/TextWorks and DB/Text *WebPublisher* are both accessing the same textbase). When publishing a “live” textbase, there are special considerations:

- DB/TextWorks (and/or other software used to maintain the textbase, such as DB/Text[®] *PowerPack*[™]) must have file access to the textbase where it resides on the HTTP server.
- Textbase record image files should be located in the same directory as the textbase, so both DB/TextWorks and DB/Text *WebPublisher* can access them. If they are not in the same directory, the full path name should be included in the Image field (for example, \\SERVER\VOL2\IMAGES\MOM.JPG) so DB/TextWorks will look in the specified location. If users specify image drives (choose **Tools>Options>Imaging**) and have a drive letter mapped to \\SERVER\VOL2\, you can omit the drive and volume.

Important! If the image will be accessed using HTTP, *WebPublisher* will strip off the path and use the location specified in DBTWPub.INI.

- Internet applications can display only GIF or JPEG images in a browser without a special plug-in. DB/TextWorks can display JPEG and many other image formats, but not GIF.
- Heavy maintenance activity may interfere with the ability of your Web clients to search the textbase. If this is a problem, schedule such activity for off-peak hours, perhaps using Deferred indexing and off-peak posting of the deferred updates.

Publishing on an Intranet

The textbase designer may have created forms that include links to launch applications. For example, a link on a report may be intended to open a document in Microsoft Word:

```
<a href="http://test/myfile.doc">Click here</a>
```

To successfully launch an application, Web clients need access to the necessary application(s) and files. The target file on the server should be accessible to all clients, either by HTTP addressing protocols or by file access protocols, such as Sun NFS or Novell NetWare. The textbase designer should be able to tell you which accessing protocol he or she used (see “Launching Applications on an Intranet” on page 61).

Important! All intranet users will have access to the files (for example, documents, spreadsheets, and so forth) to which a link has been provided. You can use textbase passwords with record-level security settings to hide records with links to documents that you do not want certain users to see. To prevent users from modifying documents, use file, directory, or user permissions using your network software or operating system.

Browser requirements are listed under “Client Requirements” on page 65. If some people in your organization are using older versions of browsers, you may want to install more recent versions.

File Locations

From time to time, the textbase designer may give you files, including textbases and HTML pages to mount on the server.

Textbase File Locations

A textbase consists of multiple files with the same name but different file extensions. For example, a textbase called *Sales* consists of SALES.ACF, SALES.BTX, SALES.DBR, SALES.DBO, SALES.DBS, SALES.IXL, SALES.OCC, SALES.SDO, SALES.TBA, and optionally, SALES.ACS (if the textbase uses record-level security), and SALE.LOG, SALES.INI, and SALES.HLP files.

If the textbase designer indicates that any textbases are linked, it is recommended that you place the linked textbases in the same directory. If they are in the same directory, omit the path from the link. If they are in different directories, you must specify the path using a UNC file name.

The DBTWPU.B.INI file controls where *WebPublisher* looks for textbases. When you installed *WebPublisher*, you had the opportunity to specify one or more textbase search paths, which were written to the [WebPublisher Defaults] section of DBTWPU.B.INI. You can also optionally indicate a specific location for each textbase by adding or modifying the [WebPublisher Textbase Paths] section in DBTWPU.B.INI manually. For example:

```
[WebPublisher Defaults]
Default-Textbase-Paths=c:\webpub\textbase\,c:\data,\\server2\tbs

[WebPublisher Textbase Paths]
cars=c:\webpub\sample\
catalog=c:\data\library\
```

If you installed the *Sample* textbases, this section is already present. Do not add the section again, but add the location to the existing one. (If you add another section, it will be ignored.)

If you have only one textbase, or if you have multiple textbases of limited size without images, you may want to put them all in the same directory (for example, the TEXTBASE subdirectory). If you have several textbases with images that you want to keep separated, or you want to locate multiple textbases on different file servers, you will probably want to edit the DBTWPU.B.INI file to specify file locations.

What happens when a client submits a query

1. *WebPublisher* checks the [WebPublisher Textbase Paths] section of DBTWPub.INI for the textbase name and looks in the specified location.
2. If the textbase name does not appear in that section or the textbase does not exist in the specified location, *WebPublisher* looks in the directories specified in the Default-Textbase-Paths.
3. If it does not find it, *WebPublisher* displays a message with a **Search** button. Clicking the button initiates a recursive subdirectory search on the directories listed in the Default-Textbase-Paths line of the [WebPublisher Defaults] section.
4. If *WebPublisher* finds the textbase in steps 2 or 3, it writes the location to the [WebPublisher Textbase Paths] section, so the search will not be necessary in the future.

HTML Search Screen Locations

A search screen is an HTML page created with DB/TextWorks, which clients will use to submit queries to *WebPublisher*. It may be either a query or menu screen. You can place HTML search screens anywhere accessible to the HTTP server. They do not have to go in any particular directory. Be sure to provide clients with a way to access the search pages (for example, provide links on your home page that jump to the HTML search pages).

Image File Locations for Forms

Use DBTWPub.INI to specify the location of the following types of images:

- **Record images.** Image file names specified in textbase records and included as picture boxes, inline images, or image links in a form.
- **Fixed images.** Logo and background images used in forms and picture boxes containing fixed images.

You can specify an image location for each textbase or put images in a default location. *WebPublisher* removes any path information specified for the image in the textbase record, then constructs a URL by pre-pending the text you specify.

Use the [WebPublisher Image Locations] section of DBTWPub.INI to specify the text *WebPublisher* should use to construct the URL for record images used in specific textbases.

Use the [WebPublisher Logo Locations] section to construct URLs for fixed images (logos and backgrounds) used with specific textbases.

In the [WebPublisher Defaults] section, use the Default-Logo-Location= and Default-Image-Location= lines to specify the text used to construct image URLs if the textbase name does not appear in either of the other sections.

In the example below, *vacation* is the name of a textbase.

```
[WebPublisher Defaults]
Default-Textbase-Paths=c:\data\,c:\webpub\textbase\
Default-Logo-Location=/mylogos
Default-Image-Location=/myimages
```

```
[WebPublisher Image Locations]
vacation=/vacimage/
```

```
[WebPublisher Logo Locations]
vacation=/mylogos/vac/
```

Note: A logo or image location that starts with a forward slash (for example, /myimages) is relative to the server root. A location without a leading “http://host” or “/” is relative to the location of the query or menu screen that submits the query.

Logo and Background Image File Locations

By default, *WebPublisher* expects to find logo and background images for forms in the location specified in DBTW PUB.INI. It expects to find logos, background images, and picture boxes for query and menu screens in /DBTW-WPD/IMAGES (the HTTP equivalent to the installation IMAGES subdirectory, such as C:\PROGRAM FILES\INMAGIC\WEBPUB\IMAGES). To display the images, *WebPublisher* constructs a URL to send to the browser. For example:

```
http://www.myco.com/dbtw-wpd/images/corpllogo.gif
```

The textbase designer can specify alternate locations using the Logos or HTML tab (choose **Tools>Form Properties>Logos/HTML** in the Form Designer and **Tools>Screen Properties>Logos/HTML** in the Menu Screen and Query Screen Designers). To override the default location, the image file name must include a forward slash (/).

Examples:

```
http://www.myco.com/logos/corpllogo.gif
/backgrnd/simple.gif
pictures/me.gif
```

A logo or image location that starts with a forward slash is relative to the server root. A location without a leading “http://host” or “/” is relative to the location of the query or menu screen that passed the query to *WebPublisher*.

Note: This functionality does not apply to textbase image names stored in records. *WebPublisher* strips all path information from image names stored in records and uses the Default-Image-Location= and [WebPublisher Image Locations] settings in DBTW PUB.INI to specify their location.

Initialization Files

DB/Text *WebPublisher* reads settings from three initialization files. All three files should reside in the Windows directory on the HTTP server where *WebPublisher* is installed.

- **DBTWPUB.INI.** Textbase and image directory settings and other options specific to *WebPublisher*.
- **INMAGIC.INI.** User options shared with DB/TextWorks.
- **DBTEXT. INI.** Date settings shared with DB/TextWorks.

The Internet user account should have Read and Write access to DBTWPUB.INI and Read access to INMAGIC.INI and DBTEXT.INI. Be sure that the Internet user account has Read access to any image or textbase directories you specify.

DBTWPUB.INI

DBTWPUB.INI contains settings used only by *WebPublisher*, not shared with DB/TextWorks. It controls where DB/Text *WebPublisher* looks for textbase and image files, as well as Americans with Disabilities Act (ADA) compliance settings, WebSetMax and WebCSSOpt settings, whether a **New Search** button appears on report pages, and whether expanded displays appear in a separate browser window.

DBTWPUB.INI is created and updated by the *WebPublisher* Setup program. It may also be updated by the *WebPublisher* program when a client submits a query (the textbase location may be written to the file). To make further changes, you can edit DBTWPUB.INI directly using a text editor.

To learn more about DBTWPUB.INI, see “Textbase File Locations” on page 69 and the *WebPublisher* READMEWP.HTM file.

INMAGIC.INI

INMAGIC.INI ensures that DB/TextWorks options are shared with DB/Text *WebPublisher*. It also controls the appearance of the *WebPublisher* attribution line used on forms and search screens.

You may need to create this file yourself, or copy it from the Windows directory on the PC where DB/TextWorks is installed to the Windows directory on the HTTP server. From time to time, the textbase designer may give you a new INMAGIC.INI file to put up on the server, or may ask you to edit INMAGIC.INI. If INMAGIC.INI is absent, default settings will be assumed.

An important setting in INMAGIC.INI is SpacedRelOps in the [Inmagic DB/TextWorks] section. It specifies whether spaces are required around search operators, such as &, /, !, and :. If Boolean and range searches do not seem to be retrieving the expected records, you may need to change this setting. As described in the *WebPublisher* READMEWP.HTM file, a setting of SpacedRelOps=1 is recommended so clients can search for dates such as 12/31/98 and URLs such as `http://www.inmagic.com` without having to use surrounding quotes.

For information about other important settings, see the *WebPublisher* READMEWP.HTM file.

DBTEXT.INI

DBTEXT.INI contains date configuration settings. Specifically, it ensures that date settings chosen when DB/TextWorks was installed are shared with DB/Text *WebPublisher*.

DBTEXT.INI is needed in the following situations:

- When the textbase contains, or users will search for, dates using non-English month names.
- When dates with 2-digit years should not always be interpreted as 19xx.
- When numeric dates would be interpreted as DD/MM/YY, rather than MM/DD/YY.

You may need to create this file yourself, or copy it from the DB/TextWorks installation directory to the Windows directory on the HTTP server. As a general rule, you should not have to make frequent edits to DBTEXT.INI. For more information about DBTEXT.INI, see the *WebPublisher* READMEWP.HTM file.

Query Sets

When a client submits a query, a set identifier is assigned and the result of the query is stored in files named <tbname>.WPD and <tbname>.WPO in the QSETS subdirectory.

The dynamically generated HTML returned to the browser contains a number of parameters, including the set name, so that the query does not have to be repeated when the client requests the next partial display or the detail on a particular record.

Sets are not retained indefinitely. To conserve disk space, set “slots” are re-used when no other slots are available, beginning with the least-recently used. You can specify the number of slots allocated, using the WebSetMax= parameter in the [WebPublisher] section of the DBTWUPUB.INI file. If there is no specification, the default is 2000 sets. See the *WebPublisher* READMEWP.HTM file for details.

```
[WebPublisher]
WebSetMax=2000
```

If a slot is re-used while a client is still browsing query results, the following error message appears: *Your current query has expired. Perform the search again.* If this happens often, perhaps due to high traffic on the site, increase the parameter value in DBTWUPUB.INI.

Chapter 6: Tutorials

You can practice publishing textbases on the Web by performing the following tutorials. These exercises are based on a fictitious organization called The Model Company, which sells high-end model cars. The company uses a textbase called *Cars*, which it publishes on the Internet.

The *Cars* textbase is included with DB/Text *WebPublisher* and can be installed by running Setup. The *Cars* textbase and associated files are installed in the SAMPLE subdirectory.

Exercise #1: Searching on the Web

In this first exercise, you will go to a Web page that was created with DB/TextWorks and do a search. The records that you find will be displayed in your browser.

Searching for Records and Displaying a Report

1. Point your Web browser to `http://hostname/dbtw-wpd/sample/carsqs.htm`, where **hostname** is your domain name or IP address. The search screen illustrated below opens. This page was created with DB/TextWorks. Each box represents a field that you can search.

The screenshot shows a Microsoft Internet Explorer window titled "Search Cars - Microsoft Internet Explorer". The address bar displays "http://localhost/dbtw-wpd/sample/carsqs.htm". The web page content includes a logo of a car with an arrow pointing to the text "The Model Company, Inc.". Below the logo is a horizontal search bar. Further down, there are three labeled input fields: "Model Type: Assembled or Kit?" with the value "kit", "Name of car: Porsche, Supra, Mustang, Ford, Mercedes, Ferrari, etc." which is empty, and "Price: Type a number or use comparisons (e.g., >200) or ranges (e.g., 100 : 200)" with the value "<200". A paragraph of instructions follows, explaining search syntax like asterisks, ampersands, and slashes. At the bottom are "Submit Query" and "Reset" buttons, and a "Help" icon. The status bar at the bottom indicates "Local intranet".

2. To find all models that come as kits (not assembled) and cost less than \$200, type `kit` in the **Model Type** box and `< 200` in the **Price** box.
3. Click the **Submit Query** button. *WebPublisher* searches the textbase, finds the records that were requested, and displays them in the browser as a formatted HTML page.

The Model Company, Inc.

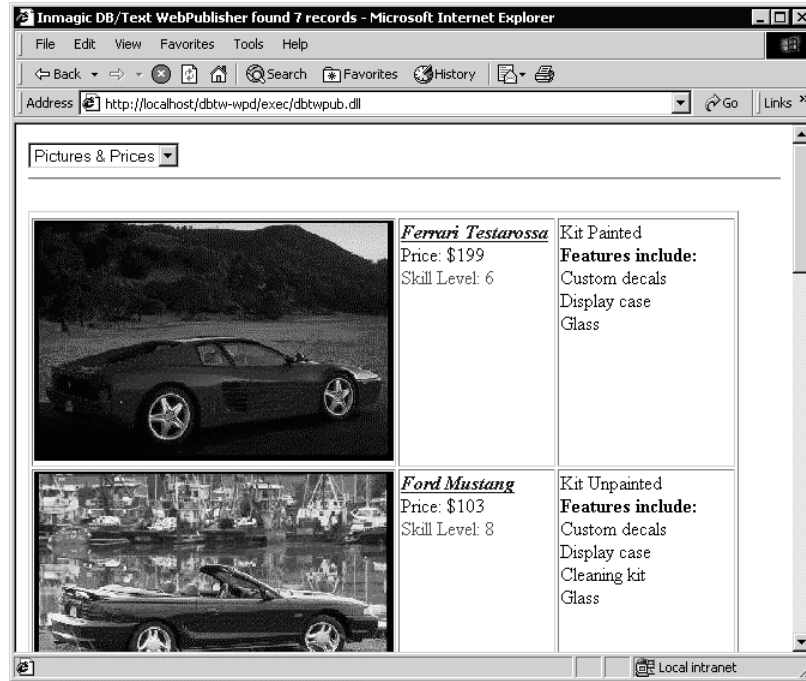
Prices

Name	Item #	Price	Available As	Colors	Status
<u>Ferrari Testarossa</u>	KP2599	\$199.00	Kit Painted	Red	In Stock
<u>Ford Mustang</u>	KU2528	\$103.00	Kit Unpainted	Red	In Stock
<u>Mercedes-Benz</u>	KP2529	\$188.00	Kit Painted	Silver, White	In Stock
<u>Nissan 300 ZX</u>	KP3528	\$199.00	Kit Painted	Red	Out of Stock
<u>Porsche</u>	KU1045	\$190.00	Kit Unpainted	White	In Stock
<u>Road Boss</u>	KU2023	\$194.00	Kit Unpainted	Yellow	In Stock
<u>Winger</u>	KU2001	\$189.00	Kit Unpainted	Red, Yellow, Blue, Hot Pink	In Stock

Prices

Use this droplist to change forms.

4. To see a different view of the records, select **Pictures & Prices** from the droplist at the top of the form. *WebPublisher* reformats the information using a form stored in the textbase.

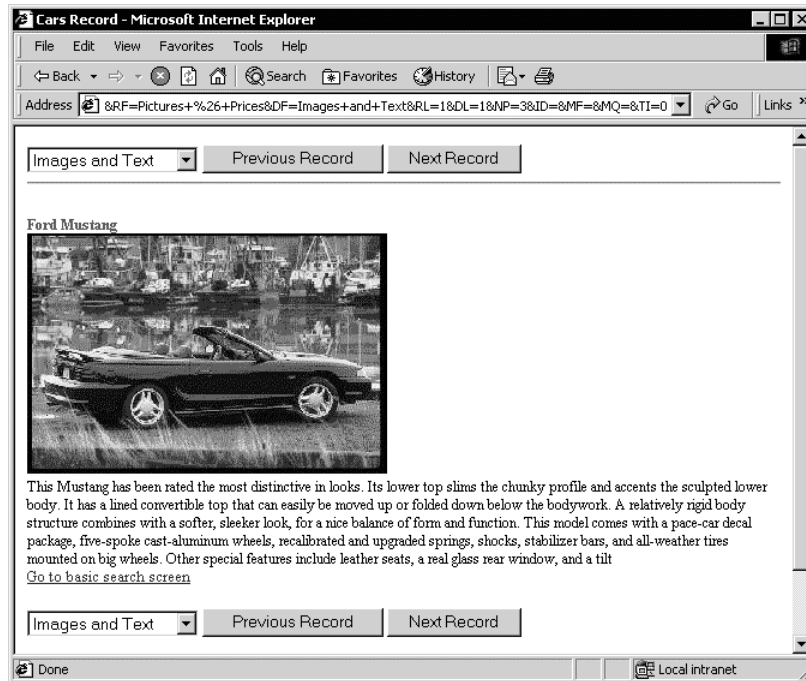


Displaying a Single Record

The reports you have seen have been summaries of the records found. You can easily “drill down” to get detailed information about individual records.

To view display records

1. Scroll to any record in the report and click the link text (the name of the car) to see detailed information about that record. When you click the link text, *WebPublisher* displays that single record using the display form specified in the query screen.



2. Click the **Next Record** and **Previous Record** buttons to see other records, one at a time.
3. Select a different form from the droplist at the top of the page to change how the record appears. This formatting is done dynamically. DB/Text *WebPublisher* gets the form stored in the textbase, applies it, and sends the formatted page back to the browser.


Exercise #2: Using the Buildware


DB/TextWorks is the buildware—the software that you use to create the textbases, forms, and search screens that you publish on the Web. In the previous exercise, you searched a textbase and displayed the results in your browser. Now you will see how to create search screens (query screens) and forms for Web use. You will also edit some records in the *Cars* textbase, to see how easy it is to keep a textbase up-to-date.

Important! For simplicity, this exercise assumes you are using a “live” textbase, which means DB/TextWorks and *WebPublisher* are accessing the same copy of the textbase. If this is not possible in your configuration, copy the sample textbase (*CARS.**) to a machine that DB/TextWorks can access. After each exercise in which you use DB/TextWorks to make changes, copy the textbase files or HTML pages back to the HTTP server so *WebPublisher* can access them.

Getting Help

If you have any questions while you are using DB/TextWorks, use the DB/TextWorks online help.

For help on a particular window, click the Help on this Window  button.

For help on a particular toolbar button or menu command, click the Help button  then click the toolbar button in question, or open a menu and click on a command.

Designing Forms

Forms determine the appearance of information retrieved from a textbase. Forms can show or hide fields, add text and punctuation, control layout and colors, and include variables and calculations appropriate for complex computations. Follow the steps below to edit the Prices form that you originally saw on the Web.

1. If you have not already done so, start DB/TextWorks.
2. Choose **File>Open** and select the *Cars* textbase, which is located by default in the *WebPublisher* SAMPLE subdirectory (for example, C:\PROGRAM FILES\INMAGIC\WEBPUB\SAMPLE\CARS).
3. Choose **Display>Design Form** to open the Open Form dialog box. Select the existing form called **Prices**, then click **OK**.
4. Change the sort order by choosing **Report Options>Compulsory Sort**.
5. Remove **Name** from the Sort Fields list by clicking the left arrow button.
6. Add **Club Price** to the Sort Fields list by selecting it in the Available Fields list and clicking the right arrow button.
7. Click **OK**.

8. To insert a hypertext link on the form, add a column to the table. Select the **Status** box, then choose **Edit>Add>Text Box**. The Text Box Properties dialog box opens.
9. On the Text tab, type **More** in the **Text** box.
10. On the HTML tab, select **Expand record link** from the **Treat text as** droplist.
11. Click **Apply**, then **Close**.
12. Remove the old expand record link from the Prices report by selecting the **Name** box and choosing **Tools>Box Properties**. On the HTML tab, select **Do not alter** from the **Treat content item as** droplist.
13. Click **Apply**, then **Close**.
14. Choose **Form Operations>Close Form Designer**, and click **Yes** when asked if you want to save the form.


Note: If you are not sharing a “live” textbase, copy the textbase files (*CARS.**) from the local machine to the **SAMPLE** subdirectory on the server now.

15. View some records in the form you just edited by pointing your browser to **CARSQS.HTM** and searching for some records. The records display in the Prices form, but are now sorted by price and there is a new column with a link for more information for each record found.

Note: The “More” link does the same thing that the car name link did in the previous exercise. However, using a separate link labeled “More,” or some other specific phrase, may provide a clearer picture as to what it does. It is up to you to decide which way works best for you and your organization.

Designing Query Screens

To enable Web clients to search a textbase, supply one or more query screens in HTML format. A query screen contains boxes in which clients type search criteria. Each box searches one or more fields. In this exercise, you will edit a query screen, export it to HTML, and then do a search using the new query screen.

1. Make sure the *Cars* textbase is still open from the last exercise.
2. Choose **Search>Design Query Screen** to open the Open Query Screen dialog box.
3. Select **Catalog QS-HTML** from the Currently Saved list. This is an existing query screen saved in the textbase. Click **OK**.
4. Add a query box by choosing **Edit>Add>Query Box**, or by clicking the Add Query Box button  on the Query Screen Designer toolbar. The Query Box Properties dialog box opens.
5. On the Fields tab, select **Features** from the Fields list and click the **Add** button.

6. On the Browse Choices tab, select the **Add Word Wheel** option button to add a Word Wheel to the screen.
7. Click **Apply**, then **Close**. The Word Wheel button will not appear until you view your page in a Web browser.
8. Choose **Tools>Screen Properties** to open the Screen Properties dialog box.
9. On the HTML tab, select the **Show Boolean operators** check box to add a Boolean droplist (**AND**, **OR**, **NOT**) to all query boxes on the page.
10. Click **OK**. The Boolean droplists will not appear until you view the page in a Web browser.
11. Choose **Screen Operations>Export Query Screen to HTML** to export the query screen to HTML. When prompted, name the file CARSQS2 .HTM and save it in any location on your Web site. Click **OK** to confirm the export.

Note: If you cannot save it directly onto your Web site, save it anywhere; then copy the HTML file to the server later.

12. Choose **Screen Operations>Close Query Screen Designer** and answer **No** when asked if you want to save the query screen.

Note: If you want to save the query screen under a different name, select **Cancel** when asked if you want to save the query screen, then choose **Screen Operations>Save Query Screen As** and choose a new name for it.

13. Point your Web browser to CARSQS2.HTM. For example, point to `http://hostname/dbtw-wpd/sample/carsqs2.htm`, where **hostname** is your domain name or IP address.

The screenshot shows a Microsoft Internet Explorer window titled "Search Cars - Microsoft Internet Explorer". The address bar displays `http://localhost/dbtw-wpd/sample/carsqs2.htm`. The main content area features a logo of a car with an arrow pointing to the text "The Model Company, Inc." followed by a horizontal row of seven empty input boxes. Below this, there are four search criteria sections, each with a dropdown menu set to "AND" and a text input field:

- Model Type: Assembled or Kit?
- Features
- Name of car: Porsche, Supra, Mustang, Ford, Mercedes, Ferrari, etc.
- Price: Type a number or use comparisons (e.g., >200) or ranges (e.g., 100 : 200)

To the right of the "Features" input field is a button labeled "Word Wheel". At the bottom of the form is a paragraph of instructions: "The Model Company designs and sells model car kits. To do a search, type words or phrases in the boxes. Use an asterisk to broaden a search to include all word endings (kit*). Use an ampersand (&) to indicate that both words must be present (porsche & supra). Use a slash (/) to indicate that either word can be present (porsche / supra). Click "Submit Query" to start the search."

The browser's status bar at the bottom shows "Done" and "Local intranet".

14. View the changes you made to the query screen:
 - A box labeled **Features** so clients can search by feature.
 - A Boolean droplist on each box so clients can choose from AND, OR, and NOT to combine search criteria among multiple fields.
 - A Word Wheel button, so clients can look at indexed information from the *Features* field in the textbase.

Editing Records

Information on a Web site is subject to change. To keep information current, DB/TextWorks offers many powerful editing features (see "Maintaining Information in a Textbase" on page 24). In the following exercise, you will use the batch modify feature to make the same change to several records at once.

1. If you have not already done so, start DB/TextWorks and open the *Cars* textbase.
2. In the **Name of car** box, type `porsche / winger / supra` and press **Enter**. This searches the *Name* field for Porsche OR Winger OR Supra.
3. On the Select Search Results Window dialog box, select **Report window**, if it is not already selected. Click **OK**, and the records matching the criteria display.

4. Choose **Records>Batch Modify** to open the Batch Modify Records in Set dialog box.
5. Select **Features** from the Field to Modify list, **Append Entry After** from the Operation choices, and **Last Entry** from the Affects choices.
6. Type `Glass windshield` in the **New Entry** box. Click **OK**.
7. On the Content Validation Mismatch dialog box, click **Accept & Update List**.
8. Click **Yes** in response to the confirmation message.
9. View the change on the Web by pointing your browser to either `CARSQS .HTM` or `CARSQS2 .HTM`.

Note: If you are not sharing a “live” textbase, close the textbase; then copy the textbase files (*CARS.**) from the local machine to the *SAMPLE* subdirectory on the server.

10. Type `porsche / winger / supra` in the **Name of car** box and click **Submit Query**. The matching records display.
11. Select **Pictures & Prices** from the form droplist and notice that the new feature, `Glass windshield`, is listed.

Now that you have a basic understanding of how to publish textbases on the Web, try creating some textbases of your own, following the instructions in the separate *Inmagic DB/TextWorks User's Manual*.

Appendix: Troubleshooting

This appendix can help you solve some common problems. For a complete list of DB/Text *WebPublisher* error messages, see the WPMMSG.HTM. file.

Message: DB/Text *WebPublisher* could not find textbase '<name>' on the server.

There are several possibilities. The textbase might not be located in any of the directory trees specified in the Default-Textbase-Paths= line in DBTWPU.B. You can open DBTWPU.B in a text editor and add the textbase location to the [WebPublisher Textbase Paths] section.

If the textbase resides on a different drive than the HTTP server, use the UNC directory name (for example, \\SERVER\VOL2\DIRECTORY), rather than the drive letter (for example, C:\DIRECTORY) in the [WebPublisher Textbase Paths] section and/or the Default-Textbase-Paths= line in DBTWPU.B.

The Internet user account (for example, IUSR_SERVERNAME) submitting the query must have Read access to the directory where the textbase resides.

Long file names with spaces can cause problems with some HTTP servers. Use short file names (eight characters) for compatibility.

Message: The textbase '<tbname>' has not yet been located on the server.

The textbase location has not been specified in the [WebPublisher Textbase Paths] section of the DBTWPU.B file or *WebPublisher* cannot find the textbase in the specified location. Click the **Search** button on the page to have *WebPublisher* do a recursive subdirectory search for the textbase in the locations specified in the Default-Textbase-Paths= line in DBTWPU.B.

Message: DB/Text *WebPublisher* found textbase '<name>' - you may re-execute your query (press Back twice).

WebPublisher successfully found the textbase and wrote its location to the [WebPublisher Textbase Paths] section of the DBTWPU.B file. If you press the **Back** button twice and perform the query again and get the message "The textbase '<tbname>' has not yet been located on the server," the Internet user account does not have Write access to DBTWPU.B. You need to either provide Write access or edit DBTWPU.B and add the textbase location.

Message: Your current query has expired. Perform the search again.

This error indicates that the slot occupied by a query set has been re-used by someone else's query. Too much time may have passed since the query was first performed, or too many users may be searching the textbase simultaneously. To increase the maximum number of query sets that can be stored, open DBTW.PUB.INI, which is located in the Windows directory on the HTTP server. Edit the **WebSetMax** parameter in the [WebPublisher] section. The default value is 2000, which is usually sufficient. If it is already at 2000 and you are still getting this error, increase the value.

```
[WebPublisher]
WebSetMax=2000
```

Symptom: Multiple JavaScript errors when loading a query or menu screen in a Web browser.

Make sure that the /DBTW-WPD virtual directory has been mapped (aliased) to the *WebPublisher* installation directory and has the necessary access (Read, Execute, and Scripts). Some HTTP servers (for example, O'Reilly WebSite) may also require explicit mapping to the /DBTW-WPD/EXEC/ virtual directory.

Symptom: Netscape 3.0 opens a Save As dialog box for DBTW-Q-MAIN.JS when loading a query or menu screen.

The following registry setting on the HTTP server is required for Netscape 3.0 to work properly with Microsoft IIS 3.0. Reboot the server after making this change:

Location:

HKEY_LOCAL_MACHINE\System\ControlSet001\Services\Inetinfo\Parameters\MimeMAP\

String value: application/x-javascript.js,,5

ValueData: ""

Symptom: Netscape 3.0 displays "garbage" in tabular reports including links.

Netscape 3 renders JavaScript within a table as visible code. Upgrade to at least version 4 of Netscape, or change the screen resolution or window size so the link fits all on one line.

Symptom: Netscape 3.0 displays "garbage" when using O'Reilly WebSite as the HTTP server.

Caching in Netscape 3.0 does not work properly with O'Reilly WebSite 1.1f. Upgrade to a newer version of O'Reilly WebSite or at least version 4 of Netscape.

Symptom: I found records, but I get an error when I try to change the form, display more records, or click link text on the report.

Verify that *WebPublisher* has Full access to the query set files (<textbase>.WPD and <textbase>.WPO) and the subdirectory where they reside (QSETS).

Symptom: When I submit a query, I do not find the intended records.

If you are doing a Boolean or range search (&, /, !, :), the problem may be related to the SpacedRelOps setting. Interpretation of search symbols is controlled by the SpacedRelOps setting in the [Inmagic DB/TextWorks] section of the INMAGIC.INI file, located in the Windows directory on the HTTP server. If SpacedRelOps=0, spaces are not required around search symbols. They are always treated as Boolean operators unless embedded within a quoted phrase. If SpacedRelOps=1, spaces are required around search symbols. They are treated as search symbols only if surrounded by spaces. The advantage of SpacedRelOps=1 is that users can search for dates such as 12/31/98 or URLs such as `http://www.inmagic.com` without having to use surrounding quotes.

Other possibilities are that the field is protected by a password, the query syntax is wrong, or Boolean operators are being misunderstood. The textbase designer can help users by adding instructions or examples to the query screen, supplying droplists for fields with limited variations, omitting Boolean buttons if the target audience is likely to be confused, and/or supplying both simple and advanced search screens.

**Symptom: The Word Wheel buttons do not appear in my browser.
Nothing happens when I click a Word Wheel button in my browser.**

The Word Wheel requires version 3.0 or later of Netscape Navigator or Microsoft Internet Explorer, or any browser that supports JavaScript/JScript 1.0 and Java JDK 1.0 or later.

Symptom: The Word Wheel dialog box is empty.

The first field searched by the box may be empty. Try selecting a different field from the **Field** droplist on the Word Wheel dialog box.

Other possibilities are that the textbase may be missing, inaccessible, or damaged; that your Web browser does not support Java and needs to be upgraded to a newer version that does; that Java is disabled in your browser settings and you need to refer to your browser's online help to learn how to enable it; or that you are using Netscape and your textbase name contains spaces.

To verify that the problem is with the Word Wheel itself (and not with the search screen or textbase), do a simple search in the same search box. If the search does not work; then the problem is not specifically with the Word Wheel.

Symptom: Some forms do not appear in the Web browser droplist.

Did you choose the correct properties when saving forms? For each form that you will use on the Web, choose **Save Form As** in the Form Designer; then save the form in the **Textbase File (Public)**. You must also select the **WebPublisher Reports** check box. Also, select either the **Report Window** (for multiple-record display) or **Record Display Window** (for single-record display) check boxes.

Symptom: Records displayed after a search look “awful” in my browser.

If a Web browser does not support tables, then tabular forms used to display search results will not appear properly. To display tables properly, use version 3.0 or later of Microsoft Internet Explorer or Netscape Navigator (version 4.0 or later preferred). If you cannot upgrade your browser, use DB/TextWorks to create non-tabular report forms.

Symptom: Records displayed after a search do not look like they did in DB/TextWorks

The form may contain Raw HTML or inline images; or you have WebCSOpt=0 in DBTWPub.INI.

Symptom: Records displayed after a search are printing on top of each other (overprinting)

The form may be using a font size greater than 10 point. Change the font size to 10 point or set WebCSOpt=0 in DBTWPub.INI. Another cause could be that the form contains Raw HTML or inline images and you have WebCSOpt=2 in DBTWPub.INI. Adjust the form to better position the box containing the Raw HTML.

Glossary

API

Application Programming Interface. *See also* ISAPI.

background image

Image file specified as the page background for a form, query screen, or menu screen used on the Web.

browser

A program used to view material prepared for the Web. Browsers can interpret URLs and HTML, and can understand several Internet protocols, such as HTTP, FTP, and Gopher. Examples of browsers are Microsoft Internet Explorer and Netscape Navigator.

CGI

Common Gateway Interface. A specification for how an HTTP server communicates with external programs. A CGI executable file provides the connection between browser software and an application. Most HTTP servers support CGI.

choices droplist

An option that you can add to a field on a query screen to show a list of indexed terms when viewed on the Web. When the HTML page is viewed in a Web browser, clients can open the droplist (also known as a drop-down list) and select an item for which to search.

Deferred indexing

DB/TextWorks indexing mode setting that keeps changes and additions to the textbase in an update queue, so they can be incorporated into the indexes at a later time.

desktop

“Desktop use” means running DB/TextWorks under Windows to access a textbase, as opposed to running a Web browser and using DB/Text *WebPublisher* to access a textbase. To use textbases on the desktop, you do not need TCP/IP or HTTP.

display form

A form that is available for display of one record at a time. *Compare* report form.

droplist

See choices droplist.

expanded display

Display of a single record at a time in a Web browser, when a user clicks on a hypertext link in a report form.

fixed text

Text added to the contents of a box in a form definition. Fixed text always appears when the form is used.

form

A layout for presentation of some or all of the information in a record.

form box

A box on a form containing field information, variables, or other information.

<HEAD> section

The top-most part of an HTML page/report, often including <TITLE>, <SCRIPT>, and <STYLE> elements.

HTML

Hypertext Markup Language. HTML is a collection of tags used to mark blocks of text and assign attributes to them, to make them appear a certain way when viewed in a Web browser.

HTML search page

See query screen.

HTTP

Hypertext Transfer Protocol, which is used for distributing hypertext documents on the Internet. HTTP is layered on top of TCP/IP.

hypertext link

The connection between one hypertext document and another. When a client clicks on a hypertext link, a jump is made to another HTML page.

index streaming

Technology that downloads only the portion of information needed, to minimize load on the server. The Word Wheel uses index-streaming technology.

initial report/display form

The form that is first used in the Web browser. The initial report form is applied when the user retrieves records. The initial display form is applied when the user expands a record by clicking a hypertext link.

inline image

An image that is displayed as part of a form, instead of as a separate page.

InmagicADVANTAGE

The software maintenance program of Inmagic, Inc. through which new features are made available to customers throughout the year as they are implemented, rather than as a once-a-year upgrade. New product releases are downloaded directly from the Inmagic Web site (www.inmagic.com). For more information, contact Inmagic, Inc. or your authorized Inmagic dealer.

Note: If you have a Web browser, you can click the URL to go to the Inmagic Web site.

Internet

A collection of computer networks connecting millions of users around the world.

intranet

A LAN (Local Area Network) or WAN (Wide Area Network) that is used in a manner similar to the Internet, using TCP/IP, Web browsers, HTML pages, and so forth.

ISAPI

Internet Server Application Program Interface specification. A high-performance alternative to Common Gateway Interface (CGI) executable files. Internet Information Services, the free Web server software provided by Microsoft, supports the ISAPI protocol. *See also* API.

Java

A platform-independent programming language that is typically used on the World Wide Web. Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

JavaScript (JScript)

A scripting language created by Netscape that can be embedded into HTML documents for use on the Web. It can also be used in form scripts in DB/TextWorks. It is unrelated to Java.

jump

See hypertext link.

lead textbase

A textbase that serves as a template when designing a query screen for use with DB/Text *WebPublisher* to search multiple textbases. The lead textbase contains the query screen definition and all of the pertinent information about the other textbases to be searched.

link

See hypertext link.

live textbase

A textbase that is in one location where both DB/TextWorks and *WebPublisher* can share it. Compare this to the alternative of maintaining the textbase locally and copying it up to the HTTP server whenever you make changes.

menu screen

A list of items, each of which initiates a pre-defined query. A pre-defined query appears on the HTML page as a regular hypertext link. Clicking on the text initiates the query.

picture box

Box on a form containing a fixed image or logo, or an image referenced in the current record.

query box

Box on a query screen, which searches one or more fields. Clients type criteria in a query box to search for records.

query screen

The query screen contains boxes into which you type words, terms, or comparison expressions. Each box represents one or more fields to be searched. Also called a search screen.

report form

A form for displaying multiple records at a time, one after another (for example, as a table). Compare display form.

script button

Button you place on a form or screen, which will perform a pre-specified action when clicked. Script buttons do not appear on *WebPublisher* forms, query screens, or menu screens.

script input box

Box you can place on a query screen, menu screen, or form that lets the user type in information to be used during the processing of a script. Script input boxes do not appear on *WebPublisher* forms, query screens, or menu screens.

search screen

See query screen.

sets box

Box added to a query screen for the purpose of combining the records in one or more previously saved sets with the current search. Sets boxes do not appear on *WebPublisher* query screens.

textbase

A collection of records containing related information.

text box

A box on a form, query screen, or menu screen containing static text.

TCP/IP

Transmission Control Protocol/Internet Protocol. TCP/IP is the basic communication protocol that is the foundation of the Internet. All other protocols (such as HTTP, FTP, and Gopher) are layered on top of TCP/IP.

UNC

Universal Naming Convention. The actual name of a drive (for example, \\SERVER\VOL2), rather than the drive letter (C:\). Recommended on networks and in other situations where drives might be remapped.

URL

Uniform (or Universal) Resource Locator. A naming, or addressing, convention used to locate a site on the World Wide Web. Example: <http://www.inmagic.com>

Word Wheel

A dialog box that can be displayed in a Web browser to show indexed words or terms in the current field. Clients can paste words and terms from the dialog box and search for them. Word Wheels make searching easier and eliminate trial-and-error searches.

Index

A

- Access control, 25
- Access rights, 1
 - for INI files, 72
 - for intranets, 68
 - passwords, 25, 44
 - setting, 7
- Accounts, 7
- Added text, 36, 62
- Adding boxes
 - password boxes, 25, 44
 - to forms, 28, 32, 62
 - to menu screens, 55, 62
 - to query screens, 43, 62, 80
- Adding choices droplists, 45, 47
- Adding hypertext links
 - to forms, 30, 34, 80
 - to query screens, 45, 58
- Adding records, 24
- Adding Word Wheels, 45, 49, 81
- Advanced HTML options, 34
- Alias, 8
- Alternate link text, 62, 64
- AND searches. *See* Boolean searches
- Applets directory, 3
- Applications, launching, 61, 68

B

- Background images
 - explained, 59
 - for forms, 34, 36, 70, 71
 - for menu screens, 71
 - for query screens, 44, 57, 71
- Batch files, 6
- Batch operations, 24, 82
- Beginning text, 36
- Boolean searches
 - adding AND-OR-NOT droplists, 44, 81, 82
 - omitting AND-OR-NOT droplists, 44
 - spaces around symbols, 72

- syntax, 66
- troubleshooting, 87
- Box Properties
 - forms, 32
 - HTML, 28, 30, 33, 60, 62
 - menu screens, 39, 56
 - query screens, 43, 44, 48, 49
- Boxes
 - adding to forms, 28, 62
 - adding to menu screens, 55, 62
 - adding to query screens, 43, 62, 80
- Browse Choices, 48, 49, 81
- Browsers. *See* Web browsers
- Buildware. *See* DB/TextWorks

C

- Cars* textbase, 13, 75
- CARSXTRA.HTM, 53
- Cascading Style Sheets, 16, 37
- CGI, 2, 6, 11, 12, 16
- Changing forms, 39, 86, 87
- Choices droplist, 18, 25, 45, 47
- Client/server, 15, 16
- Clients
 - defined, 14, 15
 - how they get help, 65
 - how they search, 18, 66
 - requirements, 1, 65
- Comparison searching, 66
- Compatibility, 1
- Convert to HTML file reference, 62
- Create field name map, 46, 50

D

- Databases. *See* Textbases
- Date searching, 66, 72, 73
- DB/Text *Import Filter*, 24
- DB/Text *PowerPack*, 68

- DB/Text *WebPublisher*
 - [WebPublisher Defaults] section, 69
 - [WebPublisher Image Locations] section, 70
 - [WebPublisher Logo Locations] section, 70
 - [WebPublisher Textbase Paths] section, 69
 - attribution line, 72
 - explained, 14
 - installing, 1
 - online help, 13, 65
 - using, 17, 67, 75
- DB/TextWorks
 - explained, 14
 - installing, 1
 - online help, 79
 - using, 79
 - versions, 1
- DBTEXT.INI, 8, 72, 73
- DBTW.PUB.INI
 - explained, 67, 72
 - image file locations, 5, 6, 70, 71
 - logos and backgrounds, 5, 71
 - query set files, 73
 - textbase file locations, 4, 69, 70
- DBTW-WPD virtual directory
 - and image locations, 5, 6, 70, 71
 - creating, 8
 - explained, 3, 67
 - picture box location, 71
- Default
 - file locations, 67, 69, 70
 - forms, 25, 38
 - image locations, 70, 71
 - sort order, 26, 31
 - textbase paths, 69, 70
 - WebSetMax value, 73
 - Word Wheel behavior, 49
- Deferred indexing, 68
- Deleting records, 24
- Desktop use, 28, 29, 32, 33, 43, 60
- Directories
 - access rights, 7, 68
 - installed files, 1, 3
 - mapping virtual, 8
 - publishing textbases, 69
- Display forms, 26, 32
- Do not alter, 34, 45
- Documentation, 13, 65
- Domain name, 2
- Droplists
 - Boolean AND-OR-NOT, 44, 81, 82

- choices droplist, 18, 25, 45, 47
- omitting from forms, 40
- selecting forms, 39

E

- Editing
 - menu screens, 58
 - query screens, 53
 - records, 24, 82
- EMail link, 35, 46
- Ending text, 36
- Enterprise Server, 11
- Error messages, 65, 85
- Executables directory, 3
- Execute Query, 20
- Exercises, 75
- Expand record link, 22, 27, 30, 31, 35, 80
- Expanded display, 26, 32
- Expired query, 73, 86
- Exporting
 - menu screens to HTML, 1, 56, 58
 - query screens to HTML, 1, 38, 52, 81

F

- Fields
 - defining, 23
 - textbase record images, 6, 60, 63, 70
- File URLs, 62, 63
- Files
 - access rights, 7, 25, 68, 72
 - documentation and samples, 13
 - HTML help pages, 65
 - HTML search screens, 70
 - images, 59, 63, 68, 70, 71
 - initialization, 72
 - installed, 1, 3
 - locations. *See* Locations
 - publishing a textbase, 17, 67, 68, 69
 - query sets, 73
 - textbase, 69
 - WP*.HTM, 65
- Filters, 24
- Fixed images, 59
- Form boxes
 - adding to a form, 28, 32
- Form Properties, 28, 33

Forms

- adding hypertext links, 34, 80
- Box Properties, 32, 60
- creating, 26, 79
- default, 25, 38
- do not appear in browser droplist, 87
- droplist in Web browser, 38, 39, 40, 45, 57
- expand record link, 22, 27, 30, 35, 80
- for printed reports, 41
- how clients change forms, 39
- initial, 38
- limiting number of records displayed, 31
- omitting form droplist, 40
- properties, 28, 33
- saving, 29, 33, 40
- search highlighting, 28
- sorting records, 26, 31
- tabular, 29
- troubleshooting, 86, 87

G

GIF, 60, 68

H

<HEAD> section, 34

Height of query boxes, 43

Help

- button on search pages, 52
- designing forms, 79
- HTML pages for clients, 65
- knowledgebase and web site, 13
- online help, 13
- README file, 3, 13

Highlighting, 28

HTML

- <HEAD> section, 34
- adding links to forms, 30, 34, 80
- adding links to query screens, 45, 58
- advanced options, 34
- Do not alter, 34, 45
- E-Mail link, 35, 46
- Expand record link, 35
- exporting menu screens to, 1, 58
- exporting query screens to, 1, 38, 52, 81
- help for clients (WP*.HTM), 65
- HTML file reference, 35
- HTML properties, forms, 33
- HTML properties, passwords, 25

HTML properties, query screens, 44

HTML query screens. *See* Query screens

HTML search screens. *See* Query screens

inline image, 36

Raw HTML, 35, 45

Screen Properties, 38, 44, 81

see also search link, 35

Treat content item as, 34, 64

Treat text as, 34, 45, 58

URL, 35, 46

Using as added text, 36

Write Report to File, 41

HTML file reference, 35, 46, 62

Use alternate link text, 62, 64

HTML properties, 33, 44, 57

HTTP

access on intranets, 62, 68

access rights, 7

mapping a virtual directory, 8

Hypertext links

adding to forms, 30, 34, 80

adding to query screens, 45, 58

jumping to images, 36, 60, 70

I

Icons, 14

IIS. *See* Internet Information Server, Internet Information Services

Image link, 60, 63

Images

adding to a menu screen, 56

file formats, 60

file locations, 3, 6, 59, 60, 63, 68, 70, 71

fixed, 59

GIF, 60, 68

hypertext links, 36, 60, 70

inline, 36, 60, 70

JPG, 60, 68

Import Filter. *See* DB/Text *Import Filter*

Importing records, 24

Index streaming, 48

Indexes, viewing while searching, 47

Indexing, 25, 49

Initial

actions, 55

elements, 55

forms, 38, 39, 44

- Initialization files
 - DBTEXT.INI, 73
 - DBTWPub.INI, 4, 5, 6, 67, 69, 72
 - INMAGIC.INI, 72
- Initialization files, 72
- Inline images, 36, 60, 63, 70
- INMAGIC.INI, 72
- Installation
 - creating a virtual directory, 8
 - installing DB/Text *WebPublisher*, 2
 - installing DB/TextWorks, 1
- Internet
 - access rights, 7
 - publishing on, 17, 67, 68
 - reaching Inmagic, Inc., 13
- Internet Explorer, 1, 68
- Internet Information Server, 10
- Internet Information Services, 1, 9
- Intranets
 - access rights, 7, 68
 - defined, 61
 - publishing on, 61, 63, 68
- IP address, 2
- ISAPI, 2, 6

J

- Java, 65, 86, 87
- JavaScript, 65, 86, 87
- JPG, 60, 68
- JSCRIPT directory, 3, 6
- Jumps. *See* Links

K

- Knowledgebase, 13

L

- LAN/WAN topology, 15
- Launching applications on an intranet, 61, 68
- Lead textbase, 46, 49, 50
- Leading logo, 33, 44, 57, 59
- Links
 - adding to forms, 30, 34, 80
 - adding to query screens, 45, 58
 - jumping to images, 36, 60, 70
- Lists
 - Boolean AND-OR-NOT, 44, 81, 82

- choices droplists for searching, 47
 - selecting forms, 39
- Live textbase, 68
- Locations
 - HTML search pages, 70
 - image files, 70, 71
 - initialization files, 72
 - intranet publishing, 61, 68
 - live textbases, 68
 - logo and background images, 71
 - query set files, 73
 - textbases, 67, 68, 69
- Logos
 - explained, 59
 - forms, 33, 70, 71
 - leading, 33, 44, 57, 59
 - menu screens, 57, 71
 - query screens, 44, 71
 - trailing, 33, 44, 57, 59

M

- Maintaining textbase records, 24, 82
- Mapping a virtual directory, 8
- Mapping fields, 46, 50
- Menu screens
 - adding images, 56
 - box properties, 39, 56
 - creating, 54
 - editing, 56, 58
 - exporting to HTML, 1, 56, 58
 - form droplist in Web browser, 57
 - HTML page locations, 70
 - HTML properties, 57
 - illustration, 19
 - Menu Screen Designer, 55
 - path settings, 56
 - saving, 56
- Messages, error, 65, 85
- Microsoft Internet Explorer, 1, 68
- Microsoft Internet Information Server, 9, 10
- Microsoft Internet Information Services, 1, 9
- Microsoft Peer Web Services, 10
- Microsoft Personal Web Server, 9
- Modifying records, 24, 82
- Multiple-textbase search, 16, 46, 49

N

- Navigation buttons, 31, 39, 40, 45, 57
- Netscape Enterprise Server, 11
- Netscape errors, 86
- Netscape Navigator, 1, 68
- Network file access, 61, 62, 68
- New Search button, 22, 39, 40, 45, 57, 72
- Next button, 31, 39, 40, 45, 57
- NOT searches. *See* Boolean searches
- Number of records, 31, 38, 45, 57

O

- O'Reilly WebSite, 12
- Operating system, 2
- Options, 72
- OR searches. *See* Boolean searches

P

- Passwords, 25, 44
- Path Settings, 56
- Peer Web Services, 10
- Permissions, 1
 - for INI files, 72
 - for intranets, 68
 - passwords, 25, 44
 - setting access rights, 7
- Personal Web Server, 9
- Phrase searching, 66
- Picture boxes
 - adding to a form, 28, 32, 60
 - adding to a menu screen, 56, 59
 - adding to a query screen, 43, 59
- Placement of navigation buttons, 31, 39, 40, 45, 57
- PowerPack. See* DB/Text *PowerPack*
- Preferences, 72
- Previous button, 31, 39, 40, 45, 57
- Printed reports, 41
- Program files, 73
- Proximity searching, 66
- Publishing textbases
 - file locations, 68, 69
 - instructions, 17, 67
 - live, 68
 - on an intranet, 61, 63, 68
 - on the desktop, 28, 29, 32, 33, 43, 60

Q

- QSETS subdirectory, 3, 8
- Queries. *See* Searching
- Query boxes
 - adding Browse Choices options, 45
 - adding to a query screen, 43
 - changing height, 43
- Query screens
 - adding boxes to, 43, 80
 - adding hypertext links, 45, 58
 - Box Properties, 43, 44, 48, 49
 - creating and editing, 53, 80
 - defined, 14, 80
 - exporting to HTML, 1, 38, 52, 81
 - HTML page locations, 70
 - HTML properties, 44
 - illustration, 18
 - initial report and display forms, 44
 - query box height, 43
 - Query Screen Designer, 42
 - saving, 43
- Query sets, 3, 55, 73

R

- Range searching, 66
- Raw HTML, 35, 45
- README file, 3, 13
- Record Display button, 38
- Record Level Security, 25
- Record separators, 34
- Records
 - adding, 23, 24
 - deleting, 24
 - displaying after a search, 76
 - editing, 24, 82
 - formatting display, 26
 - importing, 24
 - searching for, 66
 - sorting, 26, 31
 - textbase record images, 6, 60, 63, 70
- Report forms. *See* Reports
- Report Window button, 38
- Reports
 - Box Properties, 32
 - creating, 26, 29, 79
 - default, 25, 38
 - displaying records after a search, 76
 - expand record link, 22, 27, 30, 80

- how clients change forms, 39
- HTML properties, 33
- initial, 38
- limiting number of records displayed, 31
- omitting form droplist, 40
- printed, 41
- search highlighting, 28
- sorting records, 26, 31
- tabular, 29
- troubleshooting, 86, 87
- Requirements
 - client, 65
 - server, 1
 - software versions, 1
- Reset button, 52
- Rights
 - for INI files, 72
 - for intranets, 68
 - passwords, 25, 44
 - setting access rights, 7

S

- Sample textbase, 13
- Saving
 - forms, 29, 33, 40
 - menu screens, 56
 - query screens, 43
 - sets, 55
- Screen Properties
 - HTML, 38, 44, 81
 - menu screens, 56, 57, 59
 - query screens, 43, 44, 59
- Scripts, 3, 6
- Scrollable query boxes, 43
- Search highlighting, 28
- Search screens. *See* Query screens
- Searching
 - adding 'See Also' search links, 35
 - adding pre-defined search links, 54
 - Boolean AND-OR-NOT droplists, 44, 66, 81
 - creating query screens, 80
 - editing query screens, 53
 - for dates, 66, 72, 73
 - for textbases, 70
 - for URLs, 72
 - for words, phrases, terms, 25, 66
 - help for clients, 65
 - HTML page locations, 70
 - multiple textbases, 16, 46, 49

- query box height, 43
- query sets, 73
- query syntax, 66
- saving pre-defined searches, 54
- search highlighting, 28
- selecting items from a list, 47
- selecting items from a Word Wheel, 48
- troubleshooting, 87
- tutorial, 75
- See Also search links, 35
- Separator text, 36
- Servers
 - creating a virtual directory, 8
 - Microsoft Internet Information Server, 10
 - Microsoft Internet Information Services, 1, 9
 - Microsoft Peer Web Services, 10
 - Microsoft Personal Web Server, 9
 - Netscape Enterprise Server, 11
 - O'Reilly WebSite, 12
 - problems, 85
 - query sets, 73
 - requirements, 1, 2
- Sets, query, 55, 73
- Setup
 - creating a virtual directory, 8
 - installing DB/Text *WebPublisher*, 2
 - installing DB/TextWorks, 2
- Show form list on expanded display page, 40, 57
- Show form list on report page, 40, 57
- Show terms by default, 49
- Silent password, 25
- Single-record display, 26, 32
- Software versions, 1
- Sorting records, 26, 31
- SpacedRelOps setting, 66, 72, 87
- Specify number of entries, 48
- Specifying the lead textbase, 50
- Submit Query, 52
- Syntax. *See* Searching
- System requirements, 2, 65

T

- Tables (tabular forms), 29, 34, 80
- TCP/IP, 2
- Term indexing, 25, 49
- Term searching, 25, 66
- Test checklist, 64
- Testing, 64

- Text boxes
 - adding to a form, 28, 32
 - adding to a menu screen, 56
 - adding to a query screen, 43
 - creating an expand record link, 31
 - HTML properties, 58
- Text, treat as HTML, 35, 45
- Textbase
 - administrator, 14
 - box, 55
 - creating, 23
 - default forms, 25
 - default sort order, 26
 - designating as lead, 46, 49, 50
 - designer, 14
 - editing records, 24, 82
 - editing structure, 68
 - files that make up, 69
 - indexing, 25, 49
 - information, 55
 - live, 68
 - location, 68, 69
 - passwords, 25
 - publishing on the Web, 17, 67, 68, 69
 - record images, 6, 60, 63, 70
 - searching on the Web, 66
 - sort order, 26
 - supplied, 13
 - troubleshooting, 64
- Trailing logo, 33, 44, 57, 59
- Treat content item as
 - Do not alter, 34
 - Expand record link, 35
 - HTML file reference, 62
 - Image link, 36, 60
 - Inline image, 36, 60
 - See Also search link, 35
- Treat text as
 - Do not alter, 34, 45
 - EMail link, 35, 46
 - HTML file reference, 35, 46
 - HTML properties, 34, 45, 58
 - URL, 35, 46
- Troubleshooting, 64, 85
- Tutorials, 75

U

- UNC, 4, 62, 69, 85
- URLs
 - constructed by *WebPublisher*, 4, 5, 67, 68, 70, 71
 - file, 62, 63
 - for image files, 4, 5, 70
 - searching for, 72
 - treat text as URL, 35, 46
- Use alternate link text, 62, 64
- Use External File Reference, 63
- Use file:// reference for image, 63
- User file, 43

V

- Versions, software, 1
- Virtual directory
 - and image locations, 5, 6, 70, 71
 - creating, 8
 - explained, 3, 67

W

- Web browsers
 - how to get help, 65
 - requirements, 65
 - supported, 1, 65
 - tables, 29
 - using droplists, 47
 - using to search, 18, 66
 - using Word Wheels, 48
- WebExpandInSeparate parameter, 30
- Webmaster, 14, 67
- WebNewSearchButton parameter, 22
- WebPublisher*. See DB/Text *WebPublisher*
- WebSetMax parameter, 73
- WebSite, O'Reilly, 12
- Windows, 9
- Word indexing, 25, 49
- Word processing file filters, 24
- Word searching, 25, 66

Word Wheel
 adding, 45, 49, 81
 explained and illustrated, 18, 48
 files, 3
 password limitations, 25
 requirements, 65, 83
 troubleshooting, 87
WP*.HTM, 65

WPD and WPO files, 73
Write Report to File (HTML), 41

Y

Your current query has expired, 73, 86